

USER INTERFACE SPECIFICATIONS FOR THE DEFENSE INFORMATION INFRASTRUCTURE (DII)

Version 2.0

1 April 1996

Joint Interoperability and Engineering Organization
Defense Information Systems Agency

FOREWORD

The user interface specifications contained herein define the “look and feel” of information systems built in accordance with Common Operating Environment standards defined by the Defense Information Infrastructure. The specifications will be reviewed and updated by the Defense Information Systems Agency as required to remain current with technology and program requirements. This document supercedes version 1.0 of the User Interface Specifications for the Global Command and Control System (GCCS).

Comments concerning the specifications should be directed to Commanding Officer, Attn: Dr. Kathleen Fernandes, Naval Command, Control, and Ocean Surveillance Center, RDT&E Division, Code 4222, 53560 Hull Street, San Diego, CA 92152-5001, (619) 553-9224 (same for DSN), fax: (619) 553-5136, e-mail: fernande@nosc.mil.

EXECUTIVE SUMMARY

This document defines a common appearance and behavior for the user interface of information systems developed for the Defense Information Infrastructure (DII). The specifications provided here address both the “look and feel” of individual elements of the user interface as well as the design of applications and systems.

The DII is a defense-level enterprise effort to develop and field military systems that will meet the needs of the warfighter in a global information environment. The DII addresses systems in the command and control, intelligence, and combat support domains within the Department of Defense (DoD) and currently includes the Global Command and Control System and the Global Combat Support System. A user interface with a common appearance and behavior is essential to the overall usability of DII systems so that users can interact effectively with all of the software available in a system as well as generalize their experience to other systems in the DII.

The specifications in this document apply to systems installed on Hewlett Packard and Sun SPARC Series workstations and on personal computers running Windows NT. It is assumed that user interface services will be provided by X Window and Motif on UNIX-based platforms and by Microsoft Windows on PC platforms. Common Desktop Environment (CDE) will be used for desktop management on UNIX-based platforms, with the user interface “look and feel” based on the CDE version of Motif. The style guide provides an integrated set of design specifications that apply to both Motif and Windows to the extent possible, with separate specifications indicated only where the interfaces differ. Enhancements made in CDE Motif that diverge from previous versions of Motif are indicated in footnotes.

This style guide applies to the development of new software and the migration of existing software for those DII systems where the primary mode of interaction is through a graphical user interface (GUI); this document does not address the design of software providing a character-based interface or offer direction regarding possible migration of this software to a GUI-based interface. The specifications provided here conform with the native style defined for Motif and Windows, except where deviations are needed to accommodate operational requirements or constraints, provided that the deviations are consistent with established user interface guidelines. The specifications comply with direction provided by the DoD Human Computer Interface Style Guide, are consistent with military standards, handbooks, and style guides on user-computer interfaces, and incorporate guidance on user interface design published in the commercial literature.

This style guide is one of a series of related documents that define the development requirements, system architecture, and engineering for information systems in the DII. Style guide compliance is required in the development of all new software and the migration of existing software submitted for inclusion in the DII software repository. Software segments are expected to comply with all style guide specifications, with deviations occurring only when called for by operational requirements and approved by the Defense Information Systems Agency. New software shall be developed in accordance with DII requirements and be fully compliant with all style guide specifications; existing software is expected to migrate to full style compliance.

This document describes the basic elements of the user interface, explains system and application design in the DII, and addresses some additional topics in user interface design. Sections 2 and 3 describe the input devices available to users and the manner in which these devices are used to interact with the applications in a system. Sections 4, 5, and 6 provide detailed specifications describing the appearance and behavior of windows, menus and controls in the application.

Section 7 is an overview of system design with CDE and describes two models for application design in the DII. Section 8 provides design guidelines for primary, secondary, and dialog windows in the application, and section 9 presents generic guidelines concerning the presentation of information in

these windows. Sections 10 and 11 address the design of task-specific windows and availability of user support resources.

Sections 12, 13, and 14 provide guidance related to user interface internationalization, design of on-line user documentation, and user interface functionality in common support applications. The guidelines in these sections are to be implemented in the application but are not considered in determining style compliance and are not included in the checklist.

Appendix A and B identify the keyboard functions assigned to virtual keys in Motif and Windows and map these keys to the keyboards for the standard hardware platforms in the DII. Appendix C defines standard vocabulary, mnemonics, and accelerators for common actions in DII applications. Appendix D provides developer notes that translate style specifications into resource settings and configuration files for CDE, Motif, and Windows; appendix E lists the fonts for which font aliases are available in CDE. Appendix F lists acronyms and abbreviations used in the document, while Appendix G maps the terminology used in the current style guide to that in CDE Motif and Windows documentation. Appendix H identifies the style requirements for DII compliance; appendix I contains a checklist for assessing the extent to which software satisfies these requirements.

CONTENTS

1.0 INTRODUCTION	1-1
1.1 BACKGROUND	1-1
1.2 PURPOSE	1-1
1.3 ASSUMPTIONS	1-2
1.3.1 Hardware and Software Configuration	1-2
1.3.2 System Users	1-3
1.3.3 Operational Environment	1-3
1.4 SCOPE	1-3
1.5 COMPLIANCE	1-4
1.5.1 Assumptions Regarding Compliance	1-4
1.5.2 DII Compliance Requirements	1-4
1.5.2.1 Segment-Level Compliance	1-4
1.5.2.2 System-Level Compliance	1-5
1.5.2.3 Minimum Style Requirements for COE Compliance	1-5
1.5.3 Modifications to the Style Guide	1-6
1.6 DOCUMENT OVERVIEW	1-6
1.7 TYPOGRAPHIC CONVENTIONS	1-7
1.8 REFERENCES	1-7
2.0 INPUT DEVICES	2-1
2.1 POINTING DEVICE INPUT	2-1
2.1.1 The Pointer	2-1
2.1.2 Pointer Shapes	2-1
2.1.3 Pointing Device Buttons	2-3
2.2 KEYBOARD INPUT	2-4
2.2.1 Fixed Function Keys	2-4
2.2.2 Variable Function Keys	2-4
2.2.3 Text Entry	2-5
2.2.3.1 Text Entry Modes	2-5
2.2.3.2 Text Entry Actions	2-6
2.3 ALTERNATIVE INPUT DEVICES	2-6
3.0 USER-COMPUTER INTERACTION	3-1
3.1 INPUT FOCUS	3-1
3.1.1 Focus Models	3-1
3.1.2 Assigning Focus with the Pointing Device	3-1
3.1.3 Assigning Focus with the Keyboard	3-1
3.2 NAVIGATION	3-1
3.2.1 Pointing Device Navigation	3-1
3.2.2 Keyboard Navigation	3-2
3.2.2.1 The Location Cursor	3-2
3.2.2.2 Navigation in Tab Groups	3-3
3.2.2.3 Location Cursor Behavior During Navigation	3-4
3.2.2.4 Text Cursor Behavior During Navigation	3-5
3.3 SELECTION	3-6
3.3.1 Pointing Device Selection Methods	3-6
3.3.2 Keyboard Selection Methods	3-8
3.4 ACTIVATION	3-10
3.4.1 Basic Activation	3-10
3.4.2 Mnemonics and Accelerators	3-10
3.4.3 Default Activation	3-11
3.4.4 Expert Activation, Previewing, and Autorepeat (Motif Only)	3-11

3.4.5 Cancel Activation	3-11
3.5 TRANSFER	3-11
3.5.1 Drag Transfer	3-11
3.5.1.1 Drag Transfer Operations	3-11
3.5.1.2 Feedback During Drag Transfer	3-12
3.5.1.3 Performance Guidelines (Motif Only)	3-13
3.5.2 Clipboard Transfer	3-13
3.5.3 Primary Transfer (Motif Only)	3-14
3.5.4 Quick Transfer (Motif Only)	3-14
3.6 INTERACTIVE CONTROL	3-15
3.6.1 Object-Action Selection Model	3-15
3.6.2 User Control of Interaction	3-15
3.6.4 Immediate Feedback	3-15
3.6.4 Error Detection	3-16
3.6.5 Explicit Destruction	3-16
3.6.6 General Undo Capability	3-16
3.6.7 Use of Processing Modes	3-17
3.6.8 Consistency in Performing Operations	3-17
4.0 WINDOWS AND WINDOW ICONS	4-1
4.1 WINDOW MANAGEMENT	4-1
4.1.1 Window Components	4-1
4.1.1.1 Title Bar	4-1
4.1.1.2 Window Menu	4-1
4.1.1.3 Window Control Buttons	4-2
4.1.1.4 Resize Border	4-3
4.1.2 Behavior in Window Families	4-3
4.1.2.1 Parent-Child Relationships	4-3
4.1.2.2 Modes of Interaction	4-4
4.1.3 Window Management Considerations	4-5
4.1.3.1 Window Size	4-5
4.1.3.2 Window Arrangement	4-5
4.1.3.3 Window Positioning	4-5
4.1.3.4 Processing in Minimized Windows	4-6
4.1.3.5 Workspace Management (Motif Only)	4-6
4.2 WINDOW ICONS	4-6
4.2.1 Appearance	4-6
4.2.2 Behavior	4-7
5.0 MENUS	5-1
5.1 PULL-DOWN MENUS	5-1
5.1.1 Appearance	5-1
5.1.2 Behavior	5-2
5.2 POP-UP MENUS	5-3
5.2.1 Appearance	5-3
5.2.2 Behavior	5-4
5.3 TEAR-OFF MENUS (Motif Only)	5-4
5.3.1 Appearance	5-4
5.3.2 Behavior	5-5
5.4 OPTION MENUS (Motif Only)	5-6
5.4.1 Appearance	5-6
5.4.2 Behavior	5-6
5.5 MENU DESIGN GUIDELINES	5-6
5.5.1 Format of Menu Options	5-6
5.5.2 Wording of Menu Options	5-7

5.5.3	Grouping into Menus and Submenus	5-7
5.5.4	Availability of Menu Options	5-9
5.5.5	Mnemonics and Accelerators in Menus	5-9
6.0	CONTROLS	6-1
6.1	PUSH BUTTONS	6-1
6.1.1	Appearance	6-1
6.1.2	Behavior	6-2
6.2	RADIO BUTTONS	6-2
6.2.1	Appearance	6-2
6.2.2	Behavior	6-3
6.3	CHECK BUTTONS	6-3
6.3.1	Appearance	6-3
6.3.2	Behavior	6-4
6.4	LABELS	6-4
6.5	TEXT FIELDS	6-4
6.5.1	Appearance	6-4
6.5.2	Behavior	6-5
6.5.2.1	Supporting Text Entry and Manipulation	6-5
6.5.2.2	Error Checking and Correction	6-6
6.6	LIST BOXES	6-7
6.6.1	Appearance	6-7
6.6.1.1	Drop-Down List Boxes (Windows Only)	6-8
6.6.1.2	Multiple Selection List Boxes (Windows Only)	6-8
6.6.2	Behavior	6-8
6.6.2.1	Navigation and Selection	6-8
6.6.2.2	Speed and Incremental Search in Lists	6-8
6.7	SCROLL BARS	6-9
6.7.1	Appearance	6-9
6.7.2	Behavior	6-10
6.8	SCALES AND GAUGES	6-10
6.8.1	Appearance	6-10
6.8.2	Behavior	6-11
6.9	SEPARATORS	6-12
6.10	COMBINATION CONTROLS	6-12
6.10.1	Combo Boxes	6-12
6.10.2	Spin Buttons	6-13
6.11	STANDARD AND NONSTANDARD CONTROLS	6-13
6.11.1	Consistent Appearance and Behavior	6-13
6.11.2	Adapting Controls When Using Commercial Software	6-13
7.0	SYSTEM AND APPLICATION DESIGN	7-1
7.1	CDE IMPLEMENTATION IN THE DII	7-1
7.1.1	The Desktop in DII Systems	7-1
7.1.1.1	The CDE Front Panel	7-1
7.1.1.2	DII Configuration	7-1
7.1.2	Session Management	7-2
7.1.2.1	Starting and Ending a Session	7-2
7.1.2.2	DII Configuration	7-2
7.1.3	Application Management	7-2
7.1.3.1	Accessing Applications and Application Groups	7-2
7.1.3.2	DII Configuration	7-3
7.1.4	File Management	7-3
7.1.4.1	Accessing Files and Folders	7-3
7.1.4.2	DII Configuration	7-4

7.1.5	Workspace Management	7-4
7.1.5.1	Using Workspaces	7-4
7.1.5.2	DII Configuration	7-4
7.1.6	Style Management	7-4
7.1.6.1	Customizing System Style	7-4
7.1.6.2	DII Configuration	7-5
7.2	APPLICATION DESIGN IN THE DII	7-6
7.2.1	Applications Providing Centralized Task Management	7-6
7.2.1.1	Basic Implementation	7-6
7.2.1.2	Nested Implementation	7-7
7.2.2	Applications Providing Distributed Task Management	7-8
7.2.2.1	Basic Implementation	7-9
7.2.2.2	Resource Sharing Among Applications	7-11
7.2.3	Other Design Considerations	7-12
7.2.3.1	Icon Design in CDE (Motif Only)	7-12
7.2.3.2	Accessing Segments Within an Application	7-13
8.0	APPLICATION WINDOW DESIGN	8-1
8.1	PRIMARY AND SECONDARY TASK WINDOWS	8-1
8.1.1	Window Components	8-1
8.1.2	Window Design Guidelines	8-2
8.1.2.1	Window Title	8-2
8.1.2.2	Window Menu Bar	8-2
8.1.2.3	Common Menus	8-3
8.1.2.4	Arrangement of Controls	8-4
8.1.2.5	Availability of Scroll Bars and Window Panes	8-5
8.1.2.6	Arrangement of Push Buttons	8-6
8.1.2.7	Default Push Buttons	8-7
8.1.2.8	Tool Bars	8-7
8.1.2.9	Message Bar	8-9
8.1.2.10	Draggable Objects in Windows (Motif Only)	8-9
8.1.2.11	Pop-up Menus and Text Fields in Windows	8-9
8.1.2.12	Mnemonics and Accelerators in Windows	8-10
8.1.2.13	Document Windows (Windows Only)	8-10
8.2	DIALOG WINDOWS	8-10
8.2.1	Window Components	8-10
8.2.2	Window Design Guidelines	8-11
8.2.3	Message Dialogs (Motif Only)	8-12
8.2.3.1	Error Message Windows	8-12
8.2.3.2	Information Message Windows	8-12
8.2.3.3	Question Message Windows	8-12
8.2.3.4	Warning Message Windows	8-13
8.2.3.5	Working Message Windows	8-13
8.2.4	Message Dialogs (Windows Only)	8-14
8.2.4.1	Information Message Windows	8-14
8.2.4.2	Warning Message Windows	8-14
8.2.4.3	Critical Message Windows	8-15
8.2.5	Selection Dialogs (Motif Only)	8-15
8.2.5.1	Command Windows	8-15
8.2.5.2	Prompt Windows	8-16
8.2.5.3	Selection Windows	8-16
8.2.5.4	File Selection Windows	8-17
8.2.5.5	Print Windows	8-18
8.2.6	Selection Dialogs (Windows Only)	8-19
8.2.6.1	File Open and File Save Windows	8-19

8.2.6.2 Print Windows	8-20
8.3 CONSIDERATIONS IN WINDOW DESIGN	8-20
8.3.1 Selecting Controls to Match User Actions	8-20
8.3.2 Arranging Controls by Importance and Scanning Order	8-20
8.3.3 Designing for Efficiency in Task Performance	8-21
8.3.4 Minimizing the Opportunity for User Error	8-21
9.0 INFORMATION PRESENTATION	9-1
9.1 TEXT INFORMATION	9-1
9.1.1 Text Font, Size, and Style	9-1
9.1.2 Capitalization, Grammar, and Punctuation	9-1
9.1.3 Acronyms and Abbreviations	9-2
9.1.4 Formats for Date/Time and Latitude/Longitude	9-2
9.1.5 Wild Card Characters in Text Searches	9-3
9.1.6 Presenting Tabular Information	9-3
9.2 GRAPHICAL INFORMATION	9-4
9.2.1 Line Graphs and Surface Charts	9-4
9.2.2 Bar Charts and Histograms	9-6
9.2.3 Flow Charts	9-8
9.2.4 Pie Charts	9-9
9.3 TACTICAL INFORMATION	9-9
9.3.1 Tactical Symbolology	9-9
9.3.2 Coding of Tactical Information	9-10
9.4 INFORMATION CODING	9-10
9.4.1 Color	9-10
9.4.2 Flashing	9-12
9.4.3 Reverse Video	9-12
9.4.4 Size and Shape	9-12
9.4.5 Sound	9-12
9.4.6 Text Font and Style	9-12
9.5 DYNAMIC INFORMATION	9-13
10.0 TASK-SPECIFIC WINDOW DESIGN	10-1
10.1 DATA ENTRY WINDOWS	10-1
10.2 TABULAR DATA WINDOWS	10-1
10.3 LIST-TO-LIST TRANSFER WINDOWS	10-3
10.4 MAP WINDOWS	10-4
10.4.1 Map Information	10-4
10.4.2 Map Objects	10-4
10.5 GRAPHICAL SCHEDULING WINDOWS	10-6
10.5.1 Schedule Design	10-6
10.5.2 Schedule Manipulation	10-8
11.0 USER SUPPORT RESOURCES	11-1
11.1 OBJECT-LEVEL HELP	11-1
11.1.1 Message Bar Information	11-1
11.1.2 Context-Sensitive Help	11-1
11.2 WINDOW-LEVEL HELP	11-1
11.2.1 Window Design	11-1
11.2.2 Window Content	11-2
11.3 APPLICATION-LEVEL HELP	11-3
11.4 SYSTEM-LEVEL HELP	11-3
12.0 ISSUES IN USER INTERFACE INTERNATIONALIZATION	12-1
12.1 OPERATING WITH EXTENDED CHARACTER SETS	12-1

12.1.1 Character Rendering in Non-US Languages	12-1
12.1.2 Structural Rules for Character Handling	12-2
12.2 TEXT TRANSLATION	12-3
12.2.1 Creating Internationalized English Text	12-3
12.2.2 Translating Text and Messages	12-4
12.2.3 Translating Documentation	12-5
12.3 TEXT INPUT METHODS	12-5
12.3.1 Keyboards and Keyboard Input	12-5
12.3.2 Approaches to Text Entry	12-6
12.3.3 Other Text Entry Actions	12-7
12.4 INTERNATIONALIZING USER INTERFACE FEATURES	12-8
12.4.1 Text Expansion	12-8
12.4.2 Nonlinguistic Text Features	12-9
12.4.3 Data Formats	12-10
12.4.4 Graphics	12-11
12.4.5 Keyboard Interaction	12-12
12.4.6 Text Manipulation	12-13
12.4.7 Adjustments for Bidirectional Languages	12-14
12.4.8 Printing	12-14
13.0 ON-LINE USER DOCUMENTATION	13-1
13.1 DOCUMENT LIBRARY CONTENTS	13-1
13.2 TEXT-BASED DOCUMENTS	13-2
13.2.1 Elements of Text-Based Documents	13-2
13.2.2 Single-Part Documents	13-3
13.2.3 Multi-Part Documents	13-4
13.2.4 Information Presentation and Navigation	13-8
13.2.4.1 Page Size in Multi-Part Documents	13-8
13.2.4.2 Text, Tabular, and Graphic Information	13-8
13.2.4.3 Navigation Within a Document	13-9
13.3 GRAPHICS-BASED DOCUMENTS	13-10
14.0 BASIC FUNCTIONALITY IN COMMON SUPPORT APPLICATIONS	14-1
14.1 WORD PROCESSING	14-1
14.2 GRAPHICS	14-1
14.3 BRIEFING SUPPORT	14-1
14.4 MESSAGE HANDLING	14-2
14.5 IMAGERY	14-3
A. MOTIF AND WINDOWS VIRTUAL KEYS FOR KEYBOARD FUNCTIONS	A-1
B. MAPPING OF MOTIF AND WINDOWS VIRTUAL KEYS TO DII KEYBOARDS	B-1
C. ACTION VOCABULARY	C-1
D. DEVELOPER NOTES	D-1
E. FONT LIST	E-1
F. ACRONYMS AND ABBREVIATIONS	F-1
G. MOTIF AND WINDOWS TERMINOLOGY	G-1
H. STYLE REQUIREMENTS FOR COE COMPLIANCE	H-1

I. USER INTERFACE SPECIFICATIONS CHECKLIST	I-1
INDEX	J-1

LIST OF TABLES

2-1	Mapping of virtual button names to pointing device buttons in Motif and Windows	2-3
3-1	Pointing device selection methods in Motif and Windows	3-7
3-2	Keyboard selection methods in Motif	3-9
3-3	Keyboard selection methods in Windows	3-10
10-1	Pointing device selection methods in a map window	10-6
12-1	Allowances for text expansion	12-8
D-1	Color names and RGB values for color palettes in the default DII configuration	D-3
D-2	Minimum required icon set	D-8

LIST OF FIGURES

2-1	Pointer shapes in Motif and Windows	2-2
2-2	Example of soft function keys	2-5
3-1	Example location cursors in Motif	3-2
3-2	Example location cursors in Windows	3-3
3-3	Example symbol label window in Motif	3-4
4-1	Standard window components in Motif	4-1
4-2	Interaction restrictions in modeless and modal windows	4-4
4-3	Example window icon in Motif	4-6
5-1	Example pull-down menu in Windows	5-1
5-2	Examples of menu option types in Motif	5-1
5-3	Example submenu in Motif	5-2
5-4	Example pop-up menu in Motif	5-3
5-5	Example tear-off menu and menu window in Motif	5-5
5-6	Example option menu in Motif	5-6
5-7	Example of logical ordering of menu options in Windows	5-8
5-8	Organization of options in submenus	5-8
5-9	Example of an unavailable menu option in Motif	5-9
5-10	Example mnemonics and accelerators in Windows	5-9
6-1	Example push buttons in Motif	6-1
6-2	Example radio buttons in Windows	6-3
6-3	Example check buttons in Motif	6-3
6-4	Example text field in Windows	6-4
6-5	Example text field labels providing format cues	6-5
6-6	Example of automatic justification during text entry	6-6
6-7	Example list box in Motif	6-7
6-8	Example list and text entry field used in an incremental search in Motif	6-9
6-9	Components of a scroll bar	6-10

6-10	Example scale in Motif	6-11
6-11	Example gauge in Motif	6-11
6-12	Example combo box in Motif	6-12
6-13	Example drop-down combo box in Motif	6-13
6-14	Example spin button in Motif	6-14
7-1	Design of a CTM application	7-6
7-2	Launching a CTM application	7-6
7-3	Window naming conventions in a CTM application	7-7
7-4	Nested segments in a CTM application	7-8
7-5	Design of a DTM application	7-9
7-6	Launching a DTM application	7-10
7-7	Window naming conventions in a DTM application	7-10
7-8	Providing independent or shared access to support services	7-11
7-9	Options for launching a support application in a shared-resource implementation	7-12
8-1	Example primary task window in Windows	8-1
8-2	Example secondary task windows in Motif	8-2
8-3	Example arrangement of control groups in a Motif window	8-4
8-4	Example tool bar in Motif	8-8
8-5	Example error message window in Motif	8-12
8-6	Example information message window in Motif	8-12
8-7	Example question message window in Motif	8-13
8-8	Example warning message window in Motif	8-13
8-9	Example working message window in Motif	8-14
8-10	Example information message window in Windows	8-14
8-11	Example warning message window in Windows	8-15
8-12	Example critical message window in Windows	8-15
8-13	Example command window in Motif	8-16
8-14	Example prompt window in Motif	8-16
8-15	Example selection window in Motif	8-17
8-16	Example file selection window in Motif	8-18
8-17	Example print window in Motif	8-19
9-1	Example of wording style	9-1
9-2	Example of data justification within columns	9-4
9-3	Example line graphs	9-5
9-4	Example bar graphs	9-7
9-5	Example of labeling decision points in a flow chart	9-8
9-6	Example pie chart	9-9
10-1	Example data entry window in Motif	10-1
10-2	Example tabular display window in Motif	10-2
10-3	Example tabular display window with sort capability in Motif	10-3
10-4	Example list-to-list transfer window in Motif	10-4
10-5	Example map window in Motif	10-5
10-6	Example graphical scheduling window in Motif	10-7
10-7	Example task and event labels and event icons	10-8
11-1	Example help window in Motif	11-2
12-1	Forms of Arabic letter “G”	12-2
12-2	Example formation of Arabic ligature	12-3
13-1	Format for a Library Contents page	13-1

13-2	Generic structure of text-based documents	13-2
13-3	Format for a Document Title page in a single-part and multi-part document	13-3
13-4	Format for a Main Body page in a single-part document	13-4
13-5	Format for a Document Contents page in a simple multi-part document	13-5
13-6	Format for a Document Contents and Chapter Contents page in a complex multi-part document	13-6
13-7	Format for a Main Body page in a multi-part document	13-7
13-8	Format for Front and End Matter pages	13-7
13-9	Format for Briefing Contents and Slide pages in a graphics-based document	13-11

1.0 INTRODUCTION

1.1 BACKGROUND

The Defense Information Infrastructure (DII) is a defense-level enterprise effort to develop and field military systems that will meet the needs of the warfighter in a global information environment. The DII will provide “a seamless web of communications networks, computers, software, databases, applications, data, and other capabilities that meets the information processing and transport needs of DoD users in peace and in all crises, conflict, humanitarian support, and wartime roles” (DII Master Plan, page 2-1). The DII addresses systems in the command and control, intelligence, and combat support domains within the Department of Defense (DoD) and currently includes the Global Command and Control System (GCCS) and the Global Combat Support System (GCSS).

Under the DII concept, a system will be composed of software components, called segments, selected from a DII software repository and configured to meet the needs of a particular operational community. Some of the segments in the repository will offer generic functionality that can be used by multiple communities, while other segments will be targeted to particular operational requirements. To be included in the repository, segments must conform to strict standards and specifications required to support “plug and play” integration across a range of hardware platforms. When a new capability is needed in a particular mission area, segments for common functions will be assembled and customized as needed, with new software developed only to address the requirements that are unique to the mission area. Because this software is built according to the same standards and specifications as other segments, it can be added to the DII repository as another segment available for reuse.

It is critical to the overall usability of a DII system that the segments in the repository provide a user interface with a common appearance and behavior so that users can interact effectively with all of the software available in the system as well as generalize their experience to other systems in the DII. Interface standardization is particularly important as users are provided the capability to interact with a variety of complex, multi-windowed applications within a single DII system. The benefits to be gained from standardization are increased user productivity, reduced training requirements, improved system reliability, and increased efficiency in the development of individual applications as well as entire systems.

1.2 PURPOSE

This document defines a common appearance and behavior for the user interface of software components and information systems developed for the DII. The specifications provided here address both the “look and feel” of individual elements of the user interface as well as the design of applications and systems within the DII.

Implementing a common “look and feel” enables users to identify, remember, and predict the rules and organization of a system. By building consistency in the user interface, the user can develop an effective and efficient model of how the system works. According to Mayhew in Principles and Guidelines in Software User Interface Design (p. 97), a consistent user interface is one that provides:

- Consistent location of certain types of information on screens,
- Consistent syntax of commands in a command language,
- Similar execution of analogous operations in different applications,
- Consistent design of command names and abbreviations,
- Consistent grammatical form of error messages and instructions,
- Consistent design of captions and fields on forms and displays,
- Consistent dialog style for different functions, and
- Terminology consistent with the users’ existing vocabulary.

The specifications in the current document are intended to address these areas in sufficient detail so that when users see an object on the screen, they are able to recognize both the type of function they perform with the object and the means to perform the function.

While commonality in “look and feel” is an key element of usability, the concept of an application is central to the user understanding a system’s capabilities and how to interact with them. An application can be viewed as the software available to the user to perform a set of related tasks. This software is visible to the user as a collection of window families, each providing the functionality (in terms of objects and information) needed to perform a particular task. Because the DII provides a library of reusable segments from which to build applications, the window families with which the user interacts in a system may, in fact, be taken from multiple segments, each produced by a different developer organization. In addition, it is possible for applications to share the services provided by a segment when the applications perform common tasks. This flexibility contrasts with the traditional view of an application as a standalone entity that performs one or more self-contained tasks, all with software taken from a single source.

In a traditionally designed application, the developer’s view of the software is congruent with that of the user; i.e., the manner in which the developer designs the software matches the manner in which the user accesses and interacts with it. By contrast, when software is designed more flexibly (as occurs in the DII), the developer view may diverge from that of the user. For example, a segment may not be perceived by the user as an application at all, but rather as one of several tasks in the application from which the segment was launched. Furthermore, the user may view these tasks as providing a coherent set of mission-related capabilities when, in fact, the tasks are being performed by software taken from different sources. When divergences are possible, it is important that developer decisions in designing a software segment anticipate user expectations regarding its use when it is available as an application in a system. The specifications in this style guide are intended to provide the bridge between application and segment design and ensure that segment development accurately reflects the user view of the application.

As users interact with a system, they develop a mental model about its structure and behavior and how these relate to the operational tasks they have to perform. Developers can facilitate this process by providing a conceptual framework for the tasks performed by a segment that is consistent with the user’s view of these tasks when they are available in an application. This framework has to provide predictable cues from which users can identify and understand the structure and behavior of the overall system. These cues must be similar to those provided by traditional applications, yet accommodate the possibility that the application has been constructed from multiple segments produced by different developers. At a minimum, users should be able to map each task to the window family within which it is performed. Users should also be able to identify the window families in each application and correctly anticipate which application processes will be affected as they interact with windows that may be managed by different segments.

1.3 ASSUMPTIONS

1.3.1 Hardware and Software Configuration

Hardware platforms to be supported in the DII include Hewlett Packard (HP) 700 Series and Sun SPARC Series workstations, both with POSIX-compliant operating systems, as well as personal computers (PCs) running Windows NT or Windows 95. The specifications in this style guide assume that the workstation configuration for each platform includes at least one color monitor, a keyboard, and a pointing device (such as a mouse or trackball) with two or three buttons. The specifications apply to DII systems installed on HP and Sun platforms and on PCs running Windows NT; specifications for PCs running Windows 95 will be addressed in a future version of this document.

It is assumed that user interface services in DII systems are provided by X Window and Motif on UNIX-based platforms and by Microsoft Windows on PC platforms. Common Desktop Environment (CDE) is used for desktop management on UNIX-based platforms, with a user interface “look and feel” based on the CDE version of Motif. Style specifications for DII applications and systems are consistent with certification requirements set forth in documentation for the TriTeal Enterprise Desktop (TED) implementation of CDE. Because the CDE version of Motif provides significant convergence in “look and feel” with Microsoft Windows, this style guide provides an integrated set of design specifications to the extent possible, with separate specifications indicated only where the two interfaces differ.¹ In addition, enhancements made in CDE Motif that diverge from previous versions of Motif are indicated in footnotes.

1.3.2 System Users

The primary users of DII systems are expected to be operational personnel with basic software skills but no knowledge of the operating system environment or command structure. Other user groups may include system administrators, security managers, and database administrators as required by the system. It is assumed that each system will define the functionality (i.e., specific applications) that will be available to each user category and control access to these applications during system login. For example, the functions available to a system administrator may be different than those available to normal users, and the functions available to normal users may, in turn, be configurable based on the specific tasks they perform.

1.3.3 Operational Environment

It is expected that DII systems will be installed in a range of operational settings. This style guide assumes the standard environment to be an office-like workspace with normal ambient lighting and defines a default implementation for this environment. Alternative operational settings (e.g., areas with bright sunlight and significant glare, spaces where users have to remain dark adapted) will be addressed in future versions of this document.

1.4 SCOPE

This style guide applies to the development of new software and the migration of existing software for those DII systems where the primary mode of interaction is through a graphical user interface (GUI); this document does not address the design of software providing a character-based interface or offer direction regarding possible migration of this software to a GUI-based interface. The specifications provided here conform with the native style defined for Motif and Windows, except where deviations are needed to accommodate operational requirements or constraints, provided that the deviations are consistent with established user interface guidelines. This document does not provide environmental or ergonomic specifications in areas such as lighting and noise or workstation design and layout.

DoD guidance concerning user interface standardization is published in the DoD Human Computer Interface Style Guide (i.e., volume 8 of the DoD Technical Architecture Framework for Information Management). The DoD document calls for the publication of style addenda when more detailed specifications at a domain or system level are needed by DoD organizations. The specifications presented here comply with guidelines in the DoD style guide and serve as the addendum for the DII. The specifications are consistent with Section 15 on user-computer interfaces in Military Standard (MIL-

¹ This style guide provides more detailed specifications for Motif-based applications than for those in Windows. Additional coverage of Windows-based design will be provided in a future version of this document.

STD) 1472D and with Military Handbook (MIL-HDBK) 761A and incorporate guidance on user interface design published in commercial literature and available in DoD documents. The DII style guide was derived from version 1.0 of the User Interface Specifications for GCCS and represents what would have been published in the next version of that document.

1.5 COMPLIANCE

1.5.1 Assumptions Regarding Compliance

This style guide is one of a series of related documents that define development requirements, system architecture, and engineering tools for information systems in DII. Technical implementation of DII is a cooperative effort among DoD organizations, with the Defense Information Systems Agency (DISA) defining and enforcing the DII architecture. Technical requirements for building and integrating software components are described in the DII Common Operating Environment (COE) Integration and Runtime Specification (I&RTS). This document defines eight levels of compliance representing progressively deeper integration with COE software and the runtime environment. One element of compliance addresses the extent to which a COE-based system provides a consistent “look and feel.” The definition of COE compliance in the I&RTS was taken as the starting point in specifying style-related compliance requirements for DII software. The following assumptions were made in defining these requirements:

1. Because the DII includes information systems across a range of mission areas, the style guide should provide domain-level, rather than system-level, specifications for the “look and feel” of segments in the repository, with sufficient flexibility in implementation to satisfy the range of operational requirements addressed by these systems.
2. While new software is expected to be developed in accordance with DII style specifications, most existing software is likely to predate the DII initiative and will have to evolve to COE compliance. As a result, it is important that a migration path be identified that will assist DoD organizations in transitioning to the style implementation identified as the COE target. The I&RTS defines eight compliance levels which, taken together, provide a migration path for evolving software to become fully COE compliant. The DII style guide should identify the style-specific requirements at each of these levels that will support the migration process.
3. The DII style guide should describe a “look and feel” that is within the capabilities of software developers to deliver, with successive versions of the document evolving this style in accordance with DII plans. Specific direction regarding style conformance is the responsibility of DISA and will be provided for each major release of DII software. Developers may be directed to deliver a particular conformance level based on whether the software is new or evolving. Alternatively, developers may be allowed to choose a compliance area they consider to be important or timely (e.g., to leverage style changes with other planned software upgrades), with direction only to demonstrate an increase in compliance in each new software delivery.

1.5.2 DII Compliance Requirements

1.5.2.1 Segment-Level Compliance

Compliance with the specifications in the DII style guide is required in the development of all new software and the migration of existing software submitted for inclusion in the DII software repository. Software segments are expected to comply with all style guide specifications, with deviations occurring only when called for by operational requirements and approved by DISA. New software shall be developed in accordance with DII requirements and be fully compliant with all style guide specifications; existing software is expected to migrate to full COE compliance. Segments shall comply

with the intent of the specifications; the style guide does not attempt to preclude all possible inappropriate, incorrect, or unacceptable implementations.

Appendix H maps style guide specifications to each of the style-related items included in the COE compliance checklist published in the I&RTS. Segments must satisfy all of the requirements for a given style-related item in order to be considered to comply with the item. As indicated in the I&RTS, the compliance level assigned to a segment is the highest numbered level where all of the checklist items have been satisfied. To be considered COE compliant in the style area, segments must demonstrate that they have satisfied all of the style-related requirements up to and including that level. Appendix I contains a User Interface Specifications (UIS) checklist that shall serve as documentation that the segment has been evaluated and complies with all relevant style requirements.

DISA will specify the style-specific requirements to be satisfied at each COE compliance level and provide direction concerning use of the UIS checklist by DoD organizations intending to submit software to the DII repository. Organizations shall deliver a completed checklist whenever they request compliance certification by DISA. If a deviation from style compliance requirements is desired, organizations shall request a waiver from DISA; if approved, waived items can be excluded from the checklist when software is assessed for compliance. DISA shall validate the accuracy of the completed checklist against the software and determine whether to certify the software at the level requested.

1.5.2.2 System-Level Compliance

To be considered DII-compliant, DoD information systems shall conform to the system design specifications defined in this style guide. DoD organizations are encouraged to tailor the UIS checklist to reflect system-specific mission requirements and then to assess all of the software available in the system for compliance with the checklist. Organizations desiring to deliver a DII-compliant system but with operational requirements that dictate an alternate configuration shall obtain a waiver from DISA in order to diverge from the specifications presented here.

DoD organizations desiring to define system-specific user interface requirements not addressed in this document or previously waived by DISA shall do so by documenting them in an addendum to the DII style guide. The addendum shall provide detailed guidance concerning user interface features not specifically addressed within the style guide as they relate to the needs of the user community for whom the system is intended. The addendum shall maintain consistency with the user interface style defined by DII, extending the scope and content of the specifications as needed to address unique user requirements. The addendum shall be written so that it supplements, rather than duplicates, information already included in the DII style guide. Republication of the entire DII style guide, with changes to reflect system-unique requirements, is strongly discouraged.

1.5.2.3 Minimum Style Requirements for COE Compliance

While DII style specifications are intended to provide sufficient flexibility in implementation to satisfy a range of operational requirements, there are some basic elements of a Motif- or Windows-based user interface that shall not be modified in DII software. These elements are:

- The hotspot of the pointer indicates the locus of user input with the pointing device.
- The location cursor indicates the locus of user input from the keyboard.
- Only one window has input focus at any time.
- Window management operations affect only the windows in a window family.

The above elements are central to the fundamental paradigm underlying a graphical interface and are considered essential to the DII integration process at the user interface level. DII software must satisfy these minimum style requirements in order to be considered COE compliant at level 1 (see appendix H).

Requests for waivers to diverge from these requirements shall be directed to DISA and considered on a case-by-case basis.

1.5.3 Modifications to the Style Guide

The DII style guide shall be modified as needed to ensure continued compliance with Motif, Windows, and CDE style direction, maintain consistency with DoD policy and publications on user interface design, and address new user interface technologies. Revisions to the style guide shall coincide with major DII software releases and document the continued evolution of the DII user environment. Requests to modify DII style specifications shall be submitted for consideration in accordance with configuration management procedures established for the document by DISA. Requests to modify the implementation of a user interface feature in DII software shall be directed to the organization with configuration management responsibility for the software.

1.6 DOCUMENT OVERVIEW

The remainder of this document describes the basic elements of the user interface, explains system and application design in the DII, and addresses some additional topics in user interface design. Unless otherwise indicated, the specifications apply to both Motif and Windows applications. If an implementation is available in only one of the GUIs, it is identified as such and is either indented within a paragraph or included in parentheses within a sentence.

Sections 2 and 3 describe the input devices available to users and the manner in which these devices are used to interact with the applications in a system.

Sections 4, 5, and 6 provide detailed specifications describing the appearance and behavior of windows, menus and controls in an application.

Section 7 is an overview of system design with CDE and describes two models for application design in the DII.

Section 8 provides design guidelines for primary, secondary, and dialog windows in an application, and section 9 presents generic guidelines concerning the presentation of information in these windows.

Sections 10 and 11 address the design of task-specific windows and availability of user support resources.

Sections 12, 13, and 14 provide guidance related to user interface internationalization, design of on-line user documentation, and user interface functionality in common support applications. The guidelines in these sections are to be implemented in the application but are not considered in determining style compliance and are not included in the checklist.

Appendix A and B identify the keyboard functions assigned to virtual keys in Motif and Windows and map these keys to the keyboards for the standard hardware platforms in the DII.

Appendix C defines standard vocabulary, mnemonics, and accelerators for common actions in DII applications.

Appendix D provides developer notes that translate style specifications into resource settings and configuration files for CDE, Motif, and Windows; appendix E lists the fonts for which font aliases are available in CDE.

Appendix F lists acronyms and abbreviations used in the document, while Appendix G maps the terminology used in the current style guide to that in CDE Motif and Windows documentation.

Appendix H identifies the style requirements for COE compliance; appendix I contains a checklist for assessing the extent to which software satisfies these requirements.

1.7 TYPOGRAPHIC CONVENTIONS

The following typographic conventions are used in the style guide:

- a. Push button names (e.g., Save button) and actions (e.g., Cancel, OK) are capitalized as are menu titles (e.g., File) and options (e.g., Minimize, Maximize). In addition, the names of specific windows (e.g., Map Countries window) and applications (e.g., Fuel Calc application) are capitalized.
- b. The names of buttons on the pointing device are capitalized (e.g., the Select button); when the name refers to a virtual button, it is preceded by “B” (e.g., BSelect).
- c. The names of keys on a keyboard are capitalized (e.g., Return); when the name refers to a virtual key, it is shown in brackets (e.g., <Return>).
- d. Simultaneous key combinations are indicated by presenting the key names separated by a plus; for example, Ctrl+Prev means that users hold down the Ctrl key and then press the Prev key. The same meaning applies when virtual key names are presented together (e.g., <Shift><Esc>).

1.8 REFERENCES

The specifications in this document are based on the guidelines, standards, and style guides listed below.

Government Documents

Air Force Intelligence Agency. Air Force Intelligence Data Handling System (IDHS) Style Guide. Air Force Intelligence Agency, Washington, D.C., 1990.

Avery, L.W. & Bowser, S.E. (eds.) Human Factors Design Guidelines for the Army Tactical Command and Control System (ATCCS) Soldier-Machine Interface, Version 2.0. Pacific Northwest Laboratory for the U.S. Army Tactical Command and Control System Experimentation Site, Fort Lewis, WA, 1992.

Bowen, C.D. Theater Battle Management (TBM) Human Computer Interface (HCI) Specification, Version 1.2. The MITRE Corporation, Bedford, MA, 1995.

Defense Information Systems Agency. Defense Information Infrastructure (DII) Integration and Runtime Environment Specification (I/RTES), Version 2.0 (Preliminary), 1995.

Defense Information Systems Agency. Defense Information Infrastructure Master Plan, Version 3.0, 1995.

Defense Information Systems Agency. Department of Defense Technical Architecture Framework for Information Management. Volume 2: Technical Reference Model and Standards Profile Summary, Version 2.0, 1994.

Defense Information Systems Agency. Department of Defense Technical Architecture Framework for Information Management. Volume 8: Department of Defense Human Computer Interface Style Guide, Version 3.1, 1994.

Defense Information Systems Agency. User Interface Specifications for the Global Command and Control System (GCCS), Version 1.0, 1994.

Department of the Air Force, Standard Systems Center (SSC). Graphical User Interface (GUI) Standards, Volume 1, 1993.

Department of Defense. Military Handbook 761A. Human Engineering Guidelines for Management Information System. Department of Defense, Washington, D.C., September 1989.

Department of Defense. Military Standard 1472D. Human Engineering Design Criteria for Military Systems, Equipment, and Facilities. U.S. Army Missile Command, Huntsville, AL, 1989.

Department of Defense Intelligence Information Systems Management Board. Department of Defense Intelligence Information System (DODIIS) Profile of the DoD Technical Reference Model for Information Management. Bolling Air Force Base, Washington, D.C., 1993.

Naval Integrated System for Exploitation (NISE) Workstation Requirements Specification, 1991.

Operations Directorate Graphical User Interface Standards. Version 1.0. Prepared by the Joint DO/DT GUI Standards Working Group, 1994.

Smith, S.L. & Mosier, J.N. Guidelines for Designing User Interface Software (ESD0TR086-278). USAF Electronic Systems Center, Hanscom AFB, MA, 1986.

Non-Government Documents

Apple Computer, Inc. Guide to Macintosh Software Localization. Reading, MA: Addison-Wesley Publishing Co., 1992.

Apple Computer, Inc. Human Interface Guidelines: The Apple Desktop Interface. Reading, MA: Addison-Wesley Publishing Co., 1987.

CDE Documentation Group. Common Desktop Environment 1.0 User's Guide. Reading, MA: Addison-Wesley Publishing Co., 1995.

Ferguson, P.M. Motif Reference Manual for OSF/Motif Release 1.2. Sebastopol, CA: O'Reilly & Associates, Inc., 1993.

Fowler, S.L. & Stanwick, V.R. The GUI Style Guide. Boston, MA: Academic Press, Inc., 1995.

Galitz, W.O. User-Interface Screen Design. Boston, MA: QED Publishing Group, 1993.

Gardiner, M.M. & Christie, B. (eds.) Applying Cognitive Psychology to User-Interface Design. Chichester: John Wiley & Sons, 1987.

IEEE Recommended Practice for Graphical User Interface Drivability (Unapproved Draft 2), March 1992.

Kobara, S. Visual Design with OSF/Motif. Reading, MA: Addison-Wesley Publishing Co., 1991.

Mayhew, D.J. Principles and Guidelines in Software User Interface Design. Englewood Cliffs, NJ: PTR Prentice Hall, 1992.

Madell, T., Parsons, C. & Abegg, J. Developing and Localizing International Software. Englewood Cliffs, NJ: Prentice Hall, 1994.

Marcus, A., Smilonich, N. & Thompson, L. The Cross-GUI Handbook for Multiplatform User Interface Design. Reading, MA: Addison-Wesley Publishing Co., 1995.

Microsoft Corporation. The Windows Interface: An Application Design Guide. Redmond, WA: Microsoft Press, 1992.

O'Donnell, S.M. Programming for the World: A Guide to Internationalization. Englewood Cliffs, NJ: Prentice Hall, 1994.

Open Software Foundation. OSF/Motif Style Guide. Release 1.2. Englewood Cliffs, NJ: Prentice Hall, 1992.

Open Software Foundation. Motif Style Guide, Release 2.0 (Beta draft), 20 January 1994.

Open Software Foundation. OSF/Motif User's Guide. Release 1.2. Englewood Cliffs, NJ: Prentice Hall, 1992.

Root, R.W. & McFarland, A.D. Graphical User Interface Design Guidelines for Bellcore Software Products. Issue 1. Bellcore/Bell Communications Research, Piscataway, NJ, 1993.

TriTeal Corporation. TriTeal Enterprise Desktop (TED) 4.0 Advanced User's and System Administrator's Guide. Carlsbad, CA: TriTeal Corporation, 1995.

TriTeal Corporation. TriTeal Enterprise Desktop (TED) 4.0 Enhancements. Carlsbad, CA: TriTeal Corporation, 1995.

TriTeal Corporation. TriTeal Enterprise Desktop (TED) 4.0 Programmer's Guide. Carlsbad, CA: TriTeal Corporation, 1995.

TriTeal Corporation. TriTeal Enterprise Desktop (TED) 4.0 Programmer's Overview. Carlsbad, CA: TriTeal Corporation, 1995.

TriTeal Corporation. TriTeal Enterprise Desktop (TED) 4.0 Style Guide and Certification Checklist. Carlsbad, CA: TriTeal Corporation, 1995.

Uren, E., Howard, R. & Perinotti, T. Software Internationalization and Localization: An Introduction. New York: Van Nostrand Reinhold, 1993.

User Interface Design With OSF/Motif. Open Software Foundation Training Course, Version 1.2, 1992.

The specifications presented here make reference to the following additional documents:

Defense Intelligence Agency. Standard Military Graphics Symbols Manual (DIAM 65-XX) (Draft). Defense Intelligence Agency, 1990.

Department of the Army. Army Field Manual FM 101-5-1. Operational Terms and Symbols. U.S. Army Combined Arms Center, Fort Leavenworth, KS, 1985.

Department of the Army. AR 310-50. Authorized Abbreviations and Brevity Codes. Army UPDATE Publications, Washington, D.C., 1985.

Department of Defense. Military Standard 12D. Abbreviations for Use on Drawings, Specification Standards, and in Technical Documents, 1981.

Department of Defense. Military Standard 411E. Air Crew Station Alerting Systems, 1991.

Department of Defense. Military Standard 783D. Legends for Use in Air Crew Stations and on Airborne Equipment, 1984.

Department of Defense. Military Standard 2525. Common Warfighting Symbolology, Version 1, 1994.

North Atlantic Treaty Organization Standardization Agreement 2019. Military Symbols for Land-Based Systems, 1990.

NATO Standardization Agreement 4420. Display Symbolology and Colours for NATO Maritime Units, 1990.

Standards relating to the design of workstations, associated furniture, and the facilities in which they are placed can be found in MIL-STD-1472D and the following document:

American National Standards Institute. National Standard for Human Factors Engineering of Visual Display Terminal Workstations. Santa Monica, CA: The Human Factors Society, Inc., 1988.

DoD guidance on these topics can be found in the following document:

Department of Defense. DoD Human Computer Interaction (HCI) Concept Plan (Revised), 1995.

2.0 INPUT DEVICES

2.1 POINTING DEVICE INPUT

2.1.1 The Pointer

The pointing device is used to move a pointer on the screen, select objects on which the pointer is placed, and manipulate the objects directly. The pointing device is associated with a single pointer which provides a graphical representation of the location of the pointing device on the screen. The pointer may change shape depending on where it is located on the screen. The hotspot of the pointer (i.e., the active point) indicates the precise location where pointing device operations occur (e.g., the object that will be selected when users execute a select action). The location of the hotspot does not move on the screen as the pointer changes from one shape to another.

Users control the movement of the pointer by moving the pointing device; users are able to move the pointer anywhere on the screen. When users move the pointing device, the pointer moves in the corresponding direction. The position of the pointer remains where it is placed on the screen until users move it; the pointer is not moved arbitrarily by an application.

2.1.2 Pointer Shapes

The shape of the pointer depends on the functionality of the area on which the pointer is located. The pointer shapes in figure 2-1, which represent a subset of the ones defined by Motif and Windows, are to be used whenever the application provides the functionality indicated. The application can redefine the pointer shape only when the pointer is in an application window. The upper-left-pointing arrow is the general-purpose pointer for object selection in most windows. The pointer is assumed to have this shape, unless otherwise indicated.

























<u>Motif</u>	<u>Windows</u>
<u>Selection Pointers</u>  Selection pointer (used to select objects)  Sighting pointer (used to make fine position selections)	 Selection pointer (used to select objects)  Extended selection pointer (used to select lines, rows, cells)  Column selection pointer (used to select columns)
<u>Text Pointers</u>  I-beam pointer (used to manipulate text)	 I-beam pointer (used to manipulate text)
<u>Move and Resize Pointers</u>  Move pointer (indicates a move operation in progress or a resize operation before the resize direction has been determined)  Resize pointers (indicate positions and directions for area resize)	 Move pointer (indicates that the selected object is being moved)  Direction keys move pointer (indicates that direction keys will move or resize the window)  Vertical & horizontal move pointers (indicate constrained vertical & horizontal movement for the selected object)  Resize column & row pointers (indicate that a column or row can be resized)  Resize pointers (indicate that the window can be resized in the direction indicated by the pointer arrow)
<u>Working/Caution Pointers</u>  Watch pointer (indicates that an operation is being performed in a window area)  Caution pointer (indicates that an action is expected in another window area before input can be made in the current area)	 Hourglass pointer (indicates that a lengthy operation is in progress)  No-drop cursor (indicates that dropping is not allowed at this location)
<u>Help Pointers</u>  Help pointer (indicates available context-sensitive help)	 Help pointer (activates help)
<u>Other Pointers</u>  Out-of-range (indicates that the pointer has moved outside of an application area)  Menu pointer (indicates pending menu action)	 Zoom pointer (magnifies area of a window)  Split pointers (splits window horizontally or vertically)

Figure 2-1. Pointer shapes in Motif and Windows.

New pointer shapes are not created for functions that already have a shape; similarly, existing shapes are not used to symbolize functions they were not designed to represent. If a new pointer shape is created for functionality not listed in figure 2-1, it is easy to see (e.g., has high contrast with the background, does not obscure other information on the screen), with a hotspot that is obvious and easy to locate. In addition, the new shape provides a hint to its purpose and is not easily confused with other objects on the screen.

2.1.3 Pointing Device Buttons

The buttons on the pointing device are used to perform various functions related to object manipulation. Because pointing devices can have different numbers of buttons, this document uses virtual button names that identify buttons by the function they perform. BSelect (i.e., the button assigned the Select function) is used to select or activate an object, BAdjust to adjust a selection, BTransfer to transfer objects, and BMenu to display pop-up menus. This style guide assumes a pointing device with three buttons is the default in Motif applications and one with two buttons in Windows applications.

Motif Only: Two models are available for mapping pointing device buttons to virtual buttons: separate Select and Transfer functions (i.e., button 1 is used for Select and button 2 for Transfer) and integrated Select and Transfer functions (i.e., button 1 is used for both).

Table 2-1 indicates the mapping of virtual buttons to the buttons on two- and three-button pointing devices for Motif under both models and for Windows.

Table 2-1. Mapping of virtual button names to pointing device buttons in Motif and Windows.

	Motif: Integrated Select & Transfer	Motif: Separate Select & Transfer	Windows: Integrated Select & Transfer
<u>Pointing Device with Three Buttons</u>			
BSelect	Button 1	Button 1	Button 1
BAdjust	<Shift> + Button 1 (or Button 2 as an option)	<Shift> + Button 1	<Shift> + Button 1
BTransfer	Button 1 (or Button 2 as an option)	Button 2	Button 1 (or Button 2 as an option)
BMenu	Button 3	Button 3	Button 3
<u>Pointing Device with Two Buttons</u>			
BSelect	Button 1	Button 1	Button 1
BAdjust	<Shift> + Button 1	<Shift> + Button 1	<Shift> + Button 1
BTransfer	Button 1	Button 2	Button 1
BMenu	Button 2	<Alt> + Button 1	Button 2

The preferred implementation in the DII is integrated Select and Transfer functions as indicated in table 2-1.² In particular, the application supports the use of button 1 (i.e., the leftmost button on the pointing device) as BTransfer.

² Both Motif and Windows call for integrated Select and Transfer functions but allow Transfer to be assigned optionally to button 2 on the pointing device. CDE Motif gives preference to the model that integrates Select and Transfer functions while previous versions of Motif recommended the model that

Motif Only: CDE provides the option for users to assign either BAdjust or BTransfer to button 2 on a three-button pointing device when BSelect and BTransfer are integrated on button 1. When an integrated model is implemented, the application does not perform any Transfer function with button 2 when that button is used as BAdjust.

The following actions can be executed with the pointing device in both Motif and Windows:

Press	Depress and hold down a button.
Release	Release a button after it has been pressed.
Click	Press and release a button without moving the pointing device.
Double click	Press and release a button twice in rapid succession without moving the pointing device.
Move	Move the pointing device without pressing any buttons.
Drag	Move the pointing device while pressing a button.

Characteristics of the pointing device (e.g., gain and acceleration) defined at the system level (see section 7.1.6.1) are not changed by the application.

2.2 KEYBOARD INPUT

2.2.1 Fixed Function Keys

This document uses virtual keys to indicate the functions that can be executed from the keyboard. Appendix A lists the set of default functions that are defined by Motif and Windows, and appendix B maps the virtual keys for these functions to the keyboards for the standard DII hardware platforms. The application uses mappings whenever users perform any of the functions included in Appendix A.³ If the application defines additional functions, they do not conflict with key bindings listed in this appendix.

<Ctrl>, <Shift>, and <Alt> are used only to modify the function of other keys or key combinations. <Shift>+key(s) combinations are assigned to actions that extend or are complementary to the actions of the key(s) used without <Shift>. <Ctrl>+key(s) combinations are used for infrequent actions or for actions that represent larger-scale versions of the actions assigned to the unmodified key(s). <Alt> is used only to provide access to mnemonics (see sections 3.4.2 and 5.5.5). All of the function keys defined in an application are “visible” in a window (e.g., as mnemonics and accelerators) so that users do not have to rely on memory, use on-line help, or refer to system documentation in order to interact with the application from the keyboard.

2.2.2 Variable Function Keys

It is assumed that most applications will use fixed function keys (i.e., each key has only one predefined function associated with it) to execute operations from the keyboard. However, variable

separates Select and Transfer functions. The DII preference for the integrated model is a change in specifications from version 1.0 of the GCCS style guide.

³ CDE Motif recommends that all application functions be available from the keyboard. While previous versions of Motif called for keyboard support for all application functionality, they, in fact, required support for all operations except range selection in text. The DoD style guide indicates that operational military systems should provide complete interchangeability but does not define a detailed model for keyboard interaction against which to assess compliance.

function keys may also be defined for providing quick access to the operations available within the application. If variable function keys are used, the command names for the function keys are displayed in an application window in the form of soft keys, as shown in Figure 2-2. Soft keys can take on different meanings depending on the current state of the application; when the meaning of a key changes, the key labels displayed in the window are modified to reflect the action that will be executed if the key is used. If the application changes the functions assigned to a set of soft keys, it limits the functions to no more than two per key and includes an easy means (e.g., provide a “Home” key) for users to return to the set of base-level functions.

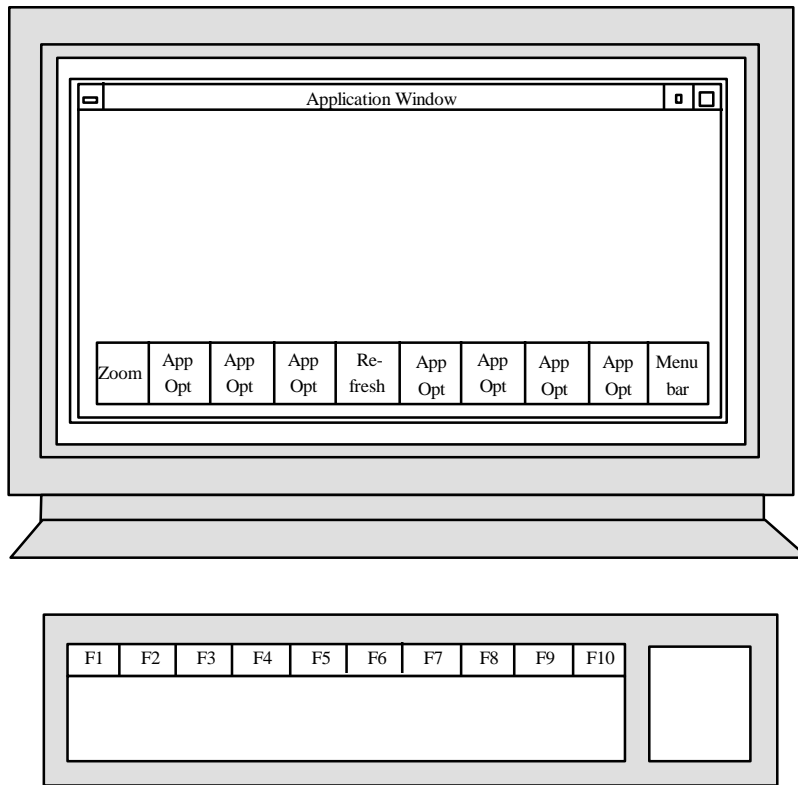


Figure 2-2. Example of soft function keys.

If variable function keys are used in the application, they are implemented in a manner consistent with input focus policy defined in section 3.1.1 (i.e., users can execute soft keys only when the window containing the keys has input focus). In addition, the actions mapped to soft keys do not conflict with the key mappings in appendix A. The keys are grayed out (to indicate their unavailability) when the window does not have focus. Developers should consider that users are likely to be working in multiple applications simultaneously, and the advantages associated with using variable function keys in the application may no longer apply when the application is one of many available to users.

2.2.3 Text Entry

2.2.3.1 Text Entry Modes

Two modes are available for text entry, with <Insert> used to toggle between the two modes. In insert mode, when users begin to type, the new character is added at the position of the text cursor. In replace mode, when users begin to type, the new character replaces the one under the text cursor.

The application provides access to both text entry modes so that users can select the mode that is more efficient given the text entry task being performed. For example, users are able to select replace mode for text entry in fields with predefined attributes (e.g., latitude/longitude and date-time group), but insert mode for free text input (e.g., the text of a message). The application does not restrict users to a single mode within a text field or arbitrarily switch between modes as users move from one field to another.

2.2.3.2 Text Entry Actions

<Space> (or <Shift><Space> in Motif) inserts a space during text entry. <Return> in Motif and <Enter> or <Ctrl><Enter> in Windows insert a carriage return in multi-line text. If no text has been selected, <Backspace> deletes the character to the left of the text cursor (backward deletion) and <Delete> deletes the character to the right of the text cursor (forward deletion). If text has been selected, both <Backspace> and <Delete> delete the selection. <Tab> inserts a tab or moves to the next tab stop in multi-line text.

Double clicking on text selects (and highlights) the word at the location of the pointer. When users highlight text and then begin typing, the text disappears, the text cursor appears, and the new text is displayed.

Motif Only: Triple clicking on text selects (and highlights) the line of text, and quadruple clicking selects (and highlights) multiple lines (e.g., a paragraph).

2.3 ALTERNATE INPUT DEVICES

Developers considering the use of a hardware configuration that includes an alternate input device other than a mouse or trackball should submit their requests to the appropriate configuration management board for approval prior to implementation. If the use of an alternate input device is approved, the manner in which users interact with the device (e.g., for navigation and selection) is consistent with the interaction models presented in section 3 of this document.

3.0 USER-COMPUTER INTERACTION

3.1 INPUT FOCUS

3.1.1 Focus Models

Two modes, explicit and implicit, are available for assigning input focus to a window. In explicit mode, focus can be moved among windows either with the pointing device or from the keyboard, and the keyboard can be used for navigation among the components in the window with focus. In implicit mode, focus moves with the pointer and cannot be controlled from the keyboard.

Regardless of the focus policy in effect, only one window on the screen has input focus at any time; within that window, only one object at a time has focus. Explicit focus is the default for the DII. However, systems may implement implicit focus or provide users with the flexibility to select either focus policy (see section 7.1.6). As a result, the application must be able to support both explicit and implicit focus (e.g., ensure that application-unique features designed for use with implicit focus also behave correctly if explicit focus is selected, and vice versa). Given that explicit focus is the default, the remainder of this document describes user interaction under this focus policy.

3.1.2 Assigning Focus with the Pointing Device

Users assign input focus by moving the pointer into a window or icon and clicking BSelect. The window with focus is indicated by a change in the color of the window frame.

3.1.3 Assigning Focus with the Keyboard

<Ctrl><Esc> displays a window listing currently running applications, and <Alt><Esc> (and <Alt><Shift><Esc> in Windows) moves focus between applications. <Alt><Tab> and <Alt><Shift><Tab> move focus forward and backward through the window families in the application, and <Alt><F6> and <Alt><Shift><F6> move focus forward and backward through the windows within a family.

Windows Only: <Ctrl><F6> and <Ctrl><Shift><F6> move focus forward and backward through the document windows in the application.

<F6> (and <Shift><F6> in Windows) move focus between panes in the active window.

3.2 NAVIGATION

3.2.1 Pointing Device Navigation

Navigation refers to how users move the pointer and location cursor on the screen. In pointing device navigation, placing the pointer on an object and clicking BSelect moves the location cursor to the object and gives it focus; the object is also selected or activated (and it is highlighted). The highlighting remains visible only while the window in which the object appears has focus. Clicking BSelect does not move focus to an object that is not selectable and cannot accept input. Holding down <Ctrl> while clicking BSelect on an object moves keyboard focus without selecting or activating the object.

If autoscrolling is available for a scrollable control such as text or a list box, placing the pointer on the control, pressing BSelect, and dragging the pointer outside the control scrolls the control in the direction of the pointer. Releasing BSelect stops autoscrolling.

3.2.2 Keyboard Navigation

3.2.2.1 The Location Cursor

The object that has focus in a window is identified by a location cursor. Only one location cursor appears in a window at any time. When focus is explicit, movement of the location cursor is controlled from the keyboard; movement of the pointing device is mapped to the pointer and does not affect the position of the location cursor on the screen.

The shape of the location cursor depends on the type of object with keyboard focus. Figures 3-1 and 3-2 show the location cursors used in Motif and Windows.

Motif Only: The box cursor is used in controls such as radio buttons, check buttons, push buttons, and text fields; the box is displayed outside the control (e.g., encloses the indicator and label in radio and check buttons). The shadow cursor⁴ is used in menus (i.e., objects whose outline is not normally shown). The item cursor is used in lists, while the fill cursor is used in very small objects such as sashes in paned windows. The text cursor is shown in the text object (e.g., text field) with keyboard focus and indicates where text typed by users will be displayed.

⁴ The [Motif Style Guide](#) refers to this cursor shape as the outline highlight location cursor. The specifications presented here refer to this shape as the shadow cursor to minimize confusion with the highlight which is the change in color (e.g., reverse video) that occurs when an object is selected.

Windows Only: The dotted box cursor is used in controls such as radio buttons, check buttons, push buttons, and list boxes; the dotted box encloses the label of the control. Highlighting is used as the location cursor in menus. When a text field initially receives focus, it highlights and the text cursor is displayed in the field. Clicking BSelect on the text removes the highlight and places the text cursor at the click location.

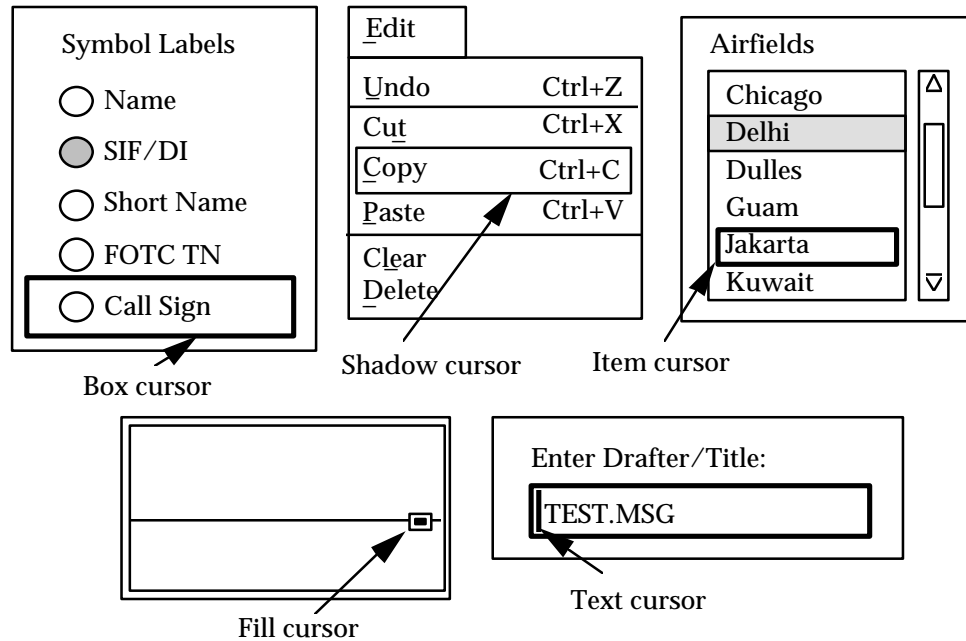


Figure 3-1. Example location cursors in Motif.

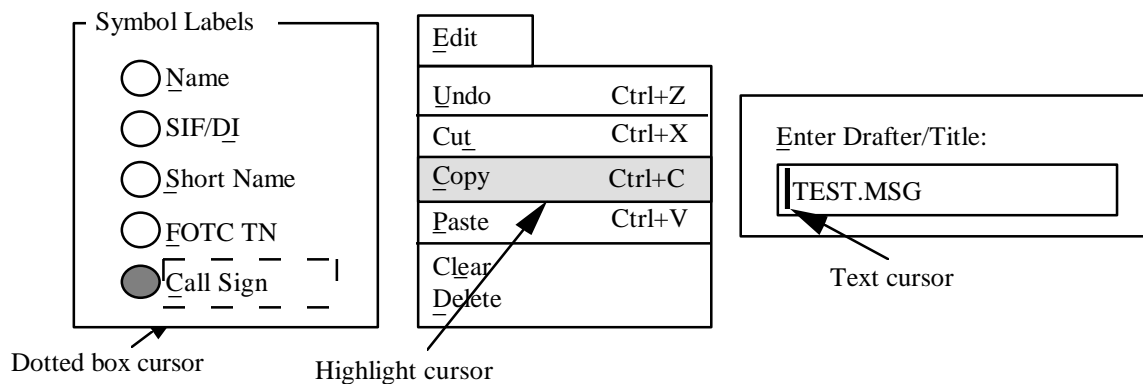


Figure 3-2. Example location cursors in Windows.

The shape of the text cursor is a vertical bar (|) in insert mode and a shaded rectangle in replace mode.⁵ The text cursor flashes when the object containing the cursor has keyboard focus. The flash rate is 2-5 Hz. The text cursor is easy to see whenever it appears in a text area. Visibility can be improved by changes in bolding, contrast, and size of the text cursor.

⁵ Previous versions of Motif use the same text cursor shape in both insert and replace mode.

The application uses existing cursor shapes to indicate keyboard focus for any new objects for which a location cursor is not already available. New cursor shapes can be created when necessary to support the functionality provided by the application.

3.2.2.2 Navigation in Tab Groups

The content of a window is divided into tab groups in order to support keyboard navigation for controls. A tab group may contain a single control, such as a list box or text field, or a set of related controls, such as a set of radio buttons, check buttons, or push buttons. Navigation keys move the location cursor between tab groups in a window and between controls within a group.

<Tab> (and <Ctrl><Tab> in Motif) moves the location cursor to the next tab group in a window, and <Shift><Tab> (and <Ctrl><Shift><Tab> in Motif) moves the cursor to the previous group. If none of the controls in a tab group can have keyboard focus (e.g., none is available for selection or activation), then <Tab> (and <Ctrl><Tab> in Motif) skips the group.

<Up>, <Down>, <Left>, and <Right> move the location cursor between the controls within the tab group with keyboard focus. The arrow keys move the location cursor one increment at a time (e.g., to the next line in text, to the next item in a list box). <Ctrl> in combination with the arrow keys increases the size of the increment (e.g., moves the text cursor to the next word rather than the next character in text). If a graphics-like object uses a positional cursor, the arrow keys move the cursor one unit (e.g., one pixel) at a time in the direction indicated by the arrow.

<Home> moves the location cursor to the leftmost element in the data, and <End> moves the cursor to the rightmost element. <Ctrl><Home> moves the location cursor to the top leftmost (i.e., beginning) element in the data, and <Ctrl><End> to the bottom rightmost (i.e., end) element. In controls such as text fields and list boxes that are scrollable, <PageUp>, <PageDown>, <Ctrl><PageUp>, and <Ctrl><PageDown> scroll the elements in the control up, down, left, or right one page minus one unit of information (e.g. one line of text) at a time, as appropriate.

Windows Only: <ScrollLock> toggles the scroll lock mode; when the mode is enabled, the navigation keys scroll data without moving the location cursor or affecting existing selections.

3.2.2.3 Location Cursor Behavior During Navigation

The application defines tab groups within each window to support efficient navigation among sets of related controls. The identification of tab groups is based on the order in which users are expected to interact with the controls in the window. This sequence is left to right, top to bottom in the window. For example, in figure 3-3, users tab from the set of radio buttons in Text Size to the text field in Text Size, then to the set of radio buttons in Symbol Size, and finally to the push buttons at the bottom of the window.

Motif Only: Text fields may be considered as individual tab groups, with <Tab> used for navigation between groups, or sets of single-line text fields may be treated as a single tab group, with the arrow keys used to navigate among them. Similarly, push buttons can be defined as individual tab groups or treated as a single tab group.⁶

Windows Only: Push buttons are defined as individual tab groups.

⁶ In previous versions of Motif, push buttons were defined as a single tab group; CDE provides the flexibility to use either <Tab> or the arrow keys to navigate within a push button group.

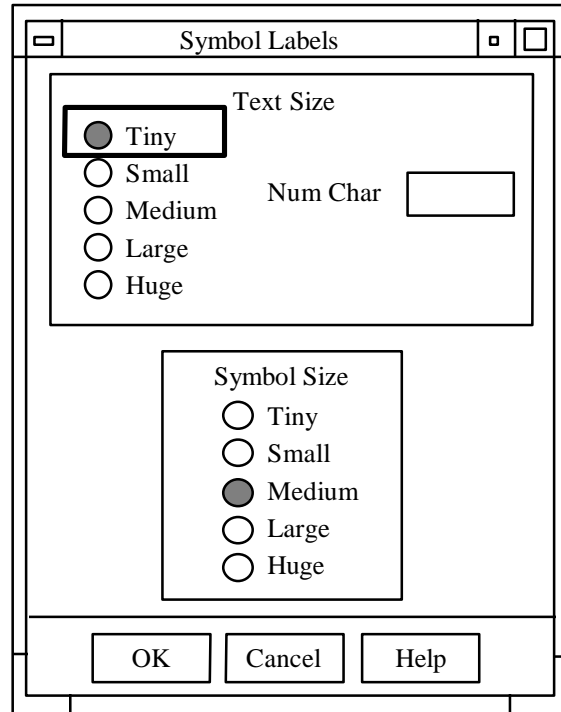


Figure 3-3. Example symbol label window in Motif.

When a window is first displayed, the location cursor is placed on the control with which users are expected to interact first. This control is usually the top leftmost one on which the location cursor can be positioned. For example, when the window shown in figure 3-3 is displayed, the location cursor is placed on the Tiny radio button in the Text Size tab group. When a window regains input focus, the location cursor is placed on the control that last had focus, provided that the control remains available for selection; otherwise, the location cursor is placed on the control users are most likely to select in the window.

The direction of cursor movement within the window is from upper left to lower right unless the object is scrollable; the location cursor wraps between the first and last tab groups in the window. When the location cursor moves into a tab group, the cursor is placed on the first available control in the group. The location cursor skips a tab group if none of the controls can have keyboard focus. In scrollable controls, keyboard focus remains on the element where it was positioned before the scrolling operation began even though the location cursor may no longer be in view. However, any keyboard action that moves the location cursor or makes a modification at the cursor location scrolls the control so that the cursor is visible.

Once the location cursor is positioned within a tab group, the arrow keys move the cursor between the available controls within the group. For example, pressing <Down> in the Symbol Size tab group moves the location cursor from Tiny to Small to Medium and so on. The location cursor is always visible as it moves within a window; i.e., there are no “invisible” tab groups which cause the location cursor to temporarily disappear as users navigate within the window. The movement of the location cursor is controlled from the keyboard; the position of the location cursor is not affected by movement of the pointer within the window. For example, the location cursor does not appear on a control within a tab group when the pointer moves into the group, and then disappear when the pointer moves out of the group.

Moving the location cursor to a control does not alter the size or position of the control. Motif and Windows differ with regard the effect of keyboard navigation on the state of the control.

Motif Only: Moving the location cursor does not change the state of the control (i.e., does not move the highlight to the control).

Windows Only: Keyboard navigation changes select state, unless Scroll Lock mode is enabled.

3.2.2.4 Text Cursor Behavior During Navigation

To assign keyboard focus to an area where text entry is possible, users move the pointer into the area and click BSelect. If the area is empty, the text cursor appears at the beginning (i.e., top leftmost part) of the area. If the area contains text, the text cursor appears between the characters that are under the pointer. If the pointer is positioned beyond the end of the text, the text cursor appears following the final text character.

<Tab> moves focus to a text field from the keyboard, highlights any text present in the field, and displays the text cursor in the field.

Motif Only: When users move the text cursor between text fields, the cursor appears at the beginning of whatever text appears in the field.

Windows Only: When users move the text cursor between text fields, the cursor appears at the end of whatever text appears in the field.

The arrow keys move the text cursor one character or one line in the direction indicated by the arrow.

The appearance of the text cursor changes when the area in which it is displayed loses focus. The text cursor disappears when the area loses focus and reappears when it regains focus. If the cursor is not removed, then it is grayed out and stops flashing when the area loses focus, and then returns to normal appearance and resumes flashing when the area regains focus. If the text cursor is removed from view when an area loses focus, the cursor re-appears at the same location when the area regains focus.

Users can place the text cursor only in areas where text entry is possible, and cannot place the cursor in areas where text entry is not possible (e.g. noneditable text fields). Text entry is possible only after the text cursor is visible at a location that can accept text entry. Text entry is not possible (i.e., is not accepted by the application) when the text cursor is not visible.

3.3 SELECTION

3.3.1 Pointing Device Selection Methods

Selection refers to the ways that users identify the elements of the interface with which they want to interact. Table 3-1 lists the methods the application is to use for selecting elements in a collection using the pointing device. In some cases, the methods vary slightly depending on whether the element is part of a one-dimensional collection such as text or a list box (i.e., where the elements can be treated as if they are in a line) or a two-dimensional collection such as a group of graphics objects (i.e., where an element has both a vertical and horizontal position within the collection). The selection methods listed in table 3-1 also apply to performing margin selection (i.e., for lines, rows, and/or columns of data) in Windows.

Motif Only: When an integrated Select/Transfer model is implemented, users can choose to map the Adjust or Transfer function (see section 2.1.3) to button 2 on the pointing device. As a result, users can perform the Adjust function in making a selection using either BSelect (as was

true with previous versions of Motif) or BAdjust (when the Adjust function is assigned to button 2). If the application uses CDE Motif, it supports both alternatives for performing selection.

Windows Only: Windows supports integrated selection and transfer using BSelect (see section 2.1.3). To distinguish between these operations, drag handles are displayed on or around an object when it is selected, and the pointer shape changes when users initiate a drag operation (see section 3.5.1 on drag transfer).

Table 3-1. Pointing device selection methods in Motif and Windows.

Single Selection:

To select one element, place the pointer on an unselected element and click BSelect. The location cursor moves to the element, it is selected, and any element that was selected is deselected.

To browse the elements in a collection, press BSelect and drag the pointer over the elements in the collection. When the pointer is on an element, it is selected, and any other element that was selected is deselected. If BSelect is released on an element, it remains selected, and the location cursor moves to the element.

Motif Only: To select multiple elements at a time, place the pointer on the first element and click BSelect or BAdjust to select it. If previously unselected, the element is selected; if previously selected, the element is deselected. In both cases, the location cursor moves to the element, and any element that was selected remain selected.

Range Selection:

To perform a range selection, press BSelect on the first element, drag the pointer until it is on the last element in the range, and release the button to complete the selection.⁷ In one-dimensional collections: The elements within the range are selected as the pointer is dragged over them. In two-dimensional collections: As the pointer is dragged through the elements, a bounding box is displayed outlining the elements being selected. When BSelect is released, the box disappears and the elements that were in the box are selected.

To extend a range selection, place the pointer on the last element to be added to the selection, then press <Shift> and click BSelect (or click BAdjust in Motif, or press <Shift><Ctrl> and click BSelect in Windows). The elements in the revised selection range (defined from the original anchor to the current pointer position in one-dimensional collections, and defined by the diagonal from the anchor to the current pointer position in two-dimensional collections) are selected, and any elements removed from the selection are deselected.

Discontiguous Selection:

To perform a discontiguous selection, place the pointer on the discontiguous element, then press <Ctrl> and click BSelect (or click BAdjust in Motif). If previously unselected, the element is selected; if previously selected, the element is deselected. The other elements in the selection remain selected. The location cursor moves to the element, and it becomes the anchor for the selection.

To extend a discontiguous selection, press <Ctrl> and BSelect (or press BAdjust in Motif) to set the anchor for the range. Drag the pointer until it is on the last element in the range, and release the button to complete the selection. In one-dimensional collections: If previously unselected, the elements in the range are selected as the pointer is dragged over them; if previously selected, the elements in the range are deselected. In two-dimensional collections: As the pointer is dragged through the elements, a bounding box is displayed outlining the elements. When BSelect (or BAdjust in Motif) is released, the box disappears. If previously unselected, the elements in the box are selected; if previously selected, the elements in the box are deselected.

⁷ In Motif, if BSelect is released before the drag threshold is exceeded, the element under the pointer is selected, and any other element that was selected is deselected. If the drag threshold is exceeded, a drag operation is initiated, and any previously selected element remains selected.

The selection method(s) available in a collection match the type of selection that users are expected to make in the collection. For example, if users are expected to select a single item in a list box, the selection method available in the list allows them to select only one item; they cannot select multiple items or deselect all items in the list. Deselection affects only the select state of the elements and does not delete any of the elements in the collection.

Windows Only: <F8> toggles Extend mode; when this mode is enabled, selection behaves as if <Shift> is locked down for all direction keys and pointing device actions. Windows supports selection modes but recommends that they be used only in special contexts. If available, they supplement standard selection methods and include a visual cue when the mode is enabled (e.g., mode information is displayed in the message bar of the window).

3.3.2 Keyboard Selection Methods

Two modes, normal and add, are available for selecting the elements in a collection using the keyboard. In normal mode, the location cursor and the highlight (indicating the current selection) move together when the arrows keys are used. In add mode, the location cursor moves independently of the highlight. The shape of the location cursor is a solid rectangle in normal mode and a dotted rectangle in add mode. <Shift><F8> toggles between the two modes. Toggling between modes does not affect the select state of objects previously selected using the pointing device.

Motif and Windows differ in their use of these modes in making selections. Table 3-2 lists the keyboard selection methods to be used in Motif applications, while table 3-3 lists the methods to be used in Windows applications.

Motif Only: Add mode is used in single and multiple selection, and normal mode in browse and range selection. Add mode may also be used in range selection. Both add and normal modes are used in discontinuous selection. Because <Space> and <Shift><Space> are mapped to text entry actions (e.g., insert a space character in text), the application is to use the alternate set of key bindings provided by Motif for performing range selection in text collections (see appendix A).

Windows Only: Normal and add mode are used in single selection, and normal mode in range selection. Both add and normal modes are used in discontinuous selection.

Table 3-2. Keyboard selection methods in Motif.

Single Selection:

To select one element, use the arrows keys to move the location cursor to an unselected element. The location cursor moves independently of the highlight. <Space> or <Select> selects the element and moves the highlight to it, and any element that was selected is deselected.

To browse the elements in a collection, use the arrow keys to move the location cursor to an element and select it; any element that was selected is deselected.

To select multiple elements one at a time, use the arrow keys to move the location cursor to an element. The location cursor moves independently of the highlight. <Space> or <Select> selects an unselected element or deselects a selected element.

Range Selection:

To perform a range selection using normal mode, use the arrow keys to move the location cursor to the first element in the range. Press <Space> or <Select> to set the anchor for the selection. The element is selected, and any element that was selected is deselected. Use the arrow keys to move the location cursor and highlight to the last element in the range. <Shift><Space> or <Shift><Select> extends the selection from the anchor to the location cursor. The location cursor remains on the last element in the range.

To perform a range selection using add mode, use the arrow keys to move the location cursor to the first element in the range. Press <Space> or <Select> to set the anchor for the selection. The element is selected, and any other element that was selected is deselected. Use the arrow keys to move the location cursor to the last element in the range; the highlight remains on the anchor element. <Shift><Space> or <Shift><Select> extends the selection from the anchor to the location cursor and selects all of the elements in the range. The location cursor remains on the last element in the range.

To extend a range selection using normal or add mode, use the arrow keys to position the location cursor on the last element in the selection. Press <Shift> and use the arrow keys to move the location cursor to the other elements to include in/remove from the selection. Elements being added to the selection highlight as the location cursor moves to them; elements being removed from the selection return to normal appearance.

Discontiguous Selection:

Select the elements in the contiguous part of the collection using normal mode (see above). Press <Shift><F8> to toggle to add mode. Select the element(s) in the discontiguous part of the collection using add mode (see above). Any elements that were selected using normal mode remain selected. Press <Shift><F8> to toggle out of add mode.

Table 3-3. Keyboard selection methods in Windows.

Single Selection:

To select a single element using normal mode, use the arrow keys to move the location cursor to an element. The highlight moves with the location cursor, and any element that was selected is deselected.

To select a single element using add mode, use the arrows keys to move the location cursor to an unselected element. The location cursor moves independently of the highlight. <Space> selects the element, moves the highlight to it, and any element that was selected is deselected.

Range Selection:

Use the arrow keys to move the location cursor to the first element in the range. The anchor for the selection is set to this element. Press <Shift> and any navigation key(s) to move the location cursor to the last element in the range. In one-dimensional collections: When these keys are pressed, the location cursor moves to the last element in the range and all elements in the range highlight. In two-dimensional collections: If the anchor element was selected, all elements in the range are selected when the keys are pressed; if the anchor element was deselected, all elements in the range are deselected.

Discontiguous Selection:

Select the elements in the contiguous part of the collection using normal mode (see above). Press <Shift><F8> to toggle to add mode. Press <Space> to toggle the select state of the element under the location cursor and set the anchor for the selection. Select the element(s) in the discontiguous part of the collection using add mode. Any elements that were previously selected remain selected. Press <Shift><F8> to toggle out of add mode.

<Backspace> and <Delete> delete the current selection.

Motif Only: <Ctrl></> is available, as appropriate, to select all of the elements in a collection and <Ctrl><\> to deselect all of the elements in a collection. In both cases, the location cursor remains at its current location.

3.4 ACTIVATION

3.4.1 Basic Activation

Activation refers to the ways users invoke the action associated with a window component or control. Placing the pointer on a button and clicking BSelect activates the button. Pressing <Space> (or <Select> in Motif) on the button with focus also activates the button. When BSelect is pressed on a button, the appearance of the button changes to indicate pending activation. This change is usually shown by highlighting the element (i.e., change in color, reverse video). If the pointer is moved off the button while BSelect is pressed, the button is not activated, and it returns to its normal appearance.

3.4.2 Mnemonics and Accelerators

A mnemonic is a single alphanumeric character in the label of a menu or control. In the window with input focus, pressing <Alt> and the mnemonic for an object moves the location cursor to the object and selects or activates it. If the location cursor is already on the object, pressing the mnemonic activates it. If the menu or control is unavailable (i.e., the label is dimmed), pressing the mnemonic has no effect (i.e., executes no action) and focus remains unchanged. An accelerator consists of one or more

keys that execute an action regardless of the position of the location cursor when the accelerator is pressed. An accelerator is executed only when the window containing the accelerator has focus. The use of mnemonics and accelerators in menus and controls is addressed in section 5.5.5 and 8.1.2.12.

3.4.3 Default Activation

When a default action is assigned to an object, the action is executed with the pointing device by double clicking BSelect on the object. If the object is used for making selections (e.g., a list box), the element under the pointer is selected and then the default action is executed. In addition, after making a selection in a window, <Enter> (and <Ctrl><Return> in Motif) invokes the default action from the keyboard. If keyboard focus is on an object other than multi-line text, <Return> in Motif also executes the default action in a window. If a default action is available, users can reverse the effects of the action (e.g., by selecting an Undo option or a Cancel push button).

3.4.4 Expert Activation, Previewing, and Autorepeat (Motif Only)

If expert activation is implemented for an object, double clicking BSelect on the object performs the regular action of the object but in a more global manner. An expert action is available only in a group of push buttons or a group of radio buttons where one of the buttons is always selected; the action is available only as a short-cut to features provided elsewhere in the window. When keyboard focus is on a button used for expert activation, there is no default action available, unless the default and expert actions are the same.

If previewing is implemented, placing the pointer on a push button or radio or check button and pressing BSelect displays information describing what happens when the button is activated; the information is removed when BSelect is released. If previewing is available, the application provides a means for users to disable it.

If autorepeat is implemented, placing the pointer on a push button and holding down BSelect executes its action repeatedly at regular intervals; autorepeat stops when BSelect is released.

3.4.5 Cancel Activation

<Esc> (or <Cancel> in Motif) cancels the action being executed and returns the object to its state prior to the action. For example, <Cancel> in the dialog window with input focus is equivalent to selecting the Cancel push button in the window.

3.5 TRANSFER

3.5.1 Drag Transfer

3.5.1.1 Drag Transfer Operations

Drag transfer allows users to move, copy, and link objects by dragging them from one location to another. The application supports these drag transfer operations.

Motif Only: Drag transfer is available for all objects that are represented as icons and for all elements that users can manipulate. Drag transfer provides redundant access to functionality available elsewhere in the window (e.g., through menus or push buttons).

As indicated in section 2.1.3, the preferred implementation in the DII is integrated Select and Transfer functions. Motif users are assumed to have a three-button pointing device and so will perform drag operations using BSelect and can also perform them using BTransfer (when the Transfer function is

assigned to button 2),⁸ while Windows users are assumed to have a two-button pointing device and so will perform drag operations using BSelect only.

Motif Only: Drag transfer is performed in the following manner:

To perform a drag move, users place the pointer on an object, then press <Shift> and drag the object using BSelect (or BTransfer). When the button is released, a copy of the object is pasted at the location of the pointer, and the original object is deleted.

To perform a drag copy, users place the pointer on an object, then press <Ctrl> and drag the object using BSelect (or BTransfer). When the button is released, a copy of the object is pasted at the location of the pointer, and the original object remains at its previous location.

To perform a drag link, users place the pointer on an object, then press <Ctrl><Shift> and drag the object using BSelect (or BTransfer). When the button is released, a link is created from the destination to the object at its original location.

The default drag operation if no modifier key is used is a move. BSelect always initiates a drag if the drag is started on a selected element; the drag starts once the drag threshold has been reached.

Windows Only: To perform a drag transfer, users place the pointer on an object and drag it to a new location using BSelect. When the button is released, a copy of the object is pasted at the location of the pointer, and the original object is deleted. How the operation is interpreted is determined by what the destination defines as the appropriate default operation. The most common default is a drag move, but the operation can also be interpreted as a drag copy or a drag link.

<Esc> (or <Cancel> in Motif) cancels a drag operation that is in progress and returns the object being dragged to its original location; releasing BSelect (or BTransfer in Motif) when not over a drop site also ends a drag operation.

Dragging a set of selected elements drags the entire selection. When a set of selected elements is moved within a collection, the elements remain selected after they have been moved.

Motif Only: Dragging an unselected element drag only the element and does not affect any other elements in the collection that may be selected. Dragging in a collection of overlapping elements occurs on the highest draggable element in the stacking order within the collection.

3.5.1.2 Feedback During Drag Transfer

During a drag operation, the pointer changes appearance (i.e., into a drag icon) to indicate that a drag is in progress.

Motif Only: The drag icon is composed of a source indicator, an operation indicator, and a state indicator.⁹ The source indicator represents the type of object(s) being dragged. The operation indicator shows the type of drag operation being performed; an operation indicator is included in the drag icon if the operation is a copy or link but not if it is a move. The state indicator has

⁸ In previous versions of Motif, the preferred implementation for drag transfer was to use BTransfer. CDE requires applications to support the use of BSelect to perform drag transfer operations.

⁹ A set of standard graphics for drag icons will be included in a future version of this document.

an arrow shape (with a hotspot for positioning) when the drag icon is on a valid drop site, and a “cannot” shape when the icon is on an invalid drop site.

Windows Only: As the pointer moves, the object, its outline, or a representation of the object moves with the pointer.

The drop site changes appearance when the drag icon is placed on it. Options in Motif include showing a solid line around the site, a raised or lowered edge around the site, or a pixmap drawn over the site. The preferred implementation for DII is a beveled edge that makes the drop site look recessed. Options in Windows include highlighting or other form of visual emphasis.

If a drag transfer is successful, the object is placed at the drop site and the drag icon is removed. If the transfer does not occur immediately, the pointer changes to a “busy” shape until the transfer is complete. If a drag transfer fails, the object remains at the drag source and the drag icon is removed.

Motif Only: The drag icon “melts into” the drop zone when dropped on a valid site and “snaps back” to the drag site when dropped on an invalid site. If the transfer fails, an error message is displayed that informs users why the failure occurred. If multiple elements are involved in a drag operation and the transfer is only partially successful, this feedback indicates which transfers succeeded and which failed. If the application provides help on a drag transfer, the dialog window providing this information contains a Cancel push button for cancelling the transfer operation.

3.5.1.3 Performance Guidelines (Motif Only)

The TED Style Guide provides the following performance guidelines with regard to executing drag transfer operations in Motif:

When BSelect is pressed, a drag transfer is initiated when the pointer has moved 10 pixels. When BTransfer is pressed, there is no drag threshold. A drag icon is displayed within 50 msec (maximum 70 msec) after the drag is initiated.

When the drag icon is moved over a drop zone, the drag icon or drop zone changes appearance within 50 msec (maximum 70 msec).

When the drag icon is dropped on a drop zone, feedback (either “melt into” or “snap back”) occurs within 50 msec (maximum 120 msec) of releasing the pointing device button. The visual effects last 200-350 msec (maximum 500 msec)

Data transfer is completed in 0.3-1.0 sec (maximum 2 sec) after the drop occurred. If the transfer takes longer than 2 sec, the pointer changes to a “ busy” shape whenever it is on the object where the transfer is occurring, and a working message window is displayed and its contents updated every 2-3 sec until the transfer is completed.

3.5.2 Clipboard Transfer

Clipboard transfer allows users to move, copy, and link objects by transferring them from their current location to a temporary clipboard and then from the clipboard to a new location. Clipboard transfer can be used to transfer text or graphics within a window and from one window to another in the same or different applications. A clipboard move consists of cut and paste actions, while a clipboard copy and a clipboard link consists of copy and paste actions.

The application supports clipboard transfer operations. These operations can be executed from a pull-down or pop-up menu or using accelerators. If the application provides access to clipboard

transfer in an Edit menu, the Cut, Copy, and Paste options in the menu execute the actions as defined in appendix C. In addition, the accelerators listed in this appendix are available for executing these actions from the keyboard whenever an object that can be edited has keyboard focus.

If the clipboard contents are text, the paste action copies the contents to the location of the text cursor, and any existing text appears to the left of the cursor. If the text has been previously selected when a paste action occurs, the selected text is deleted and the clipboard contents pasted at that location. If the clipboard contents are graphic, the paste action copies the contents to the pointer location in the window with input focus. If the paste action is invoked from the pop-up menu for an object, the clipboard contents are pasted at the insertion point in the object. If the menu is popped up over a selection, the selection is deleted and the clipboard contents replace it if possible. Pasting an object from the clipboard does not select the object. The object that was pasted remains in the clipboard until another object is cut or copied into it. Users are able to view the contents of the clipboard and are informed (e.g., in a message window) when they attempt to cut or copy an object whose size exceeds clipboard capacity.

3.5.3 Primary Transfer (Motif Only)

Primary transfer allows users to transfer a selectable object (e.g., editable text) directly to a destination without using the clipboard for intermediate storage. Primary transfer is performed in the following manner:

To perform a primary move, users select an object, move the pointer to the destination, and press <Shift> and click BTransfer. A copy of the object is pasted at the location of the pointer, and the original object is deleted.

To perform a primary copy, users select an object, move the pointer to the destination, and press <Ctrl> and click BTransfer. A copy of the object is pasted at the location of the pointer, and the original object remains at its previous location.

To perform a primary link, users select an object, move the pointer to the destination, and press <Ctrl><Shift> and click BTransfer. A link is created from the location of the pointer to the object at its original location.

In addition, if access to primary transfer is provided in an Edit menu, the Primary Copy and Primary Move options execute the actions and use the accelerators defined in appendix C.

Transferring an object by performing a primary copy or a primary link does not select the object; however, transferring an object via a primary move does. When BTransfer is used to perform a primary transfer, the default is a copy operation.

3.5.4 Quick Transfer (Motif Only)

Quick transfer allows users to temporarily select an object and immediately transfer it to a new location. Whereas primary transfer is available for editable objects, quick transfer can be used to transfer static text (e.g., the label for a text field or a push button) or graphics that are not normally selectable. Quick transfer is available in text components in the application and performed in the following manner:

To perform a quick move, users place the pointer on an object, press <Alt><Shift>, and drag the object using BTransfer. The object is temporarily selected and, when the button is released, moved to the new location.

To perform a quick copy, users place the pointer on an object, press <Alt><Ctrl>, and drag the object using BTransfer. The object is temporarily selected and, when the button is released, copied to the new location.

To perform a quick link, users place the pointer on an object, press <Alt><Ctrl><Shift>, and drag the object using BTransfer. The object is temporarily selected and, when the button is released, linked to the new location.

When <Alt> and BTransfer are used to perform a quick transfer, the default is a copy operation. When a quick transfer is performed, the object being transferred is not selected when it is displayed at the destination location.

3.6 INTERACTIVE CONTROL

3.6.1 Object-Action Selection Model

User interactions with the application are based on an object-action selection model in which users first select an object (so that it has keyboard focus) and then perform an action on that object. More than one object may be selected, and the objects may be controls (e.g., radio or check buttons), text (e.g., individual characters, strings of characters), or graphic (e.g., track symbols on a map). When users make or change a selection, no other action is performed on the selected objects. Users are informed (e.g., with feedback in the message bar of a window) when an application requires interactions that diverge from this model.

3.6.2 User Control of Interaction

The application executes an action only in response to explicit user input. Users control the pace of the interaction with the application and are not forced to interact with the application at a specified rate. However, the pace of user input does not slow down the speed of application processing or execution.

3.6.3 Immediate Feedback

Users receive an immediate indication that their action has been accepted by the application. Some visible response (e.g., a button is highlighted, the pointer changes shape) is made within 0.2 sec of any user action. When a user's request takes longer than 2-3 sec but less than 10 sec to process, the application changes the pointer shape to "busy" (e.g., a watch or hourglass). When a user's request takes longer than 10 sec to process, the application displays a working message window.¹⁰ Error feedback (e.g., when a user attempts an action that is invalid) is provided to users within 2 sec of the time the error was detected. These guidelines should be considered as recommended response times; it is expected that actual response times will be determined by factors such as the hardware configuration being used, the size of the track database being maintained, and the amount of communications processing being performed.

Users are informed when they can and cannot take an action. Visual cues are provided indicating when the application can accept input (e.g., the pointer appears as a shape supporting selection), when it is temporarily unavailable (e.g., the pointer appears as a watch or hourglass), and when it is unavailable during extended processing (e.g., a working window appears on the screen). In addition, the appearance of the objects displayed by the application provides an indication of their availability. For example, menu options provide feedback concerning the actions that can be performed on selected

¹⁰ CDE recommends that a progress indicator be displayed after 10 sec; Windows recommends the indicator be displayed after 5 sec.

object(s). If a menu option is not appropriate for an object that has been selected or if a menu option cannot be executed for some reason, users are unable to select the menu item and a visual cue (such as graying the option) is provided.

When an operation requires several actions to complete, users are prompted with the actions that need to be performed. For example, when users select a Zoom Area option from a pull-down menu, the application provides feedback explaining how to select the area to be zoomed (e.g., provides a message at the bottom of the window).

The application ignores user actions made during periods when input cannot be accepted. For example, the application accepts keystrokes made by users only when the text cursor appears in the window with input focus; if the cursor is not present in the window, the keystrokes are ignored by the application. The application disables the pointing device and/or keyboard when input may have destructive effects (e.g., when the watch or caution pointer is displayed, indicating that the application is temporarily unavailable or that input cannot be accepted at the location of the pointer). Disabling is particularly important so that any input made by users is not stored and then executed when the application becomes available again. Although an application does not allow users to override disabling, users are able to stop a process if desired (e.g., by activating a Cancel push button).

3.6.4 Error Detection

If users attempt to execute an invalid action, the application does not execute the action except to display a message describing the action that is invalid. If users make multiple errors within a single action, they are notified of each occurrence of an error. The feedback is immediate (i.e., occurs within two seconds of error detection), is visual and/or auditory, and explains the nature of the error made. If users repeat an error, the feedback is different (e.g., auditory with diagnostics or help also provided) to show users that their attempted corrective action was, in fact, processed. Users are required to correct only the invalid action and not to repeat the entire action sequence. After correcting the error, users can execute the same action (e.g., activate a push button) for re-entry that was used for the original entry.

3.6.5 Explicit Destruction

Users are required to confirm a destructive action (i.e., an action with irreversible negative consequences such as deleting a data file) before the action is executed by the application. Users do not have to confirm the action when they close a window unless the action will cause significant data loss. In general, closing a primary window is a potentially destructive action because this type of window usually generates data that should not be lost as a result of such an action. Closing a dialog window is usually nondestructive because these windows are used for obtaining additional user input that does not have to be saved.

When a confirmation prompt is displayed, the window in which the destructive action was taken remains displayed until users make a selection to confirm the action. The window is not removed when the prompt is presented and then redisplayed if users fail to confirm the destructive action. If a destructive action applies to more than one object (e.g., multiple files), users are provided with a list of the objects and allowed to select the one(s) to which the action should be applied.

3.6.6 General Undo Capability

Users are able to undo the action or selection most recently made unless the selection was one requiring explicit destruction. For example, users are able to undo actions such as cut, copy, and paste, deselect objects, return an object to its prior state before an action was executed, and retrieve information that was removed from the screen. If an undo capability cannot be provided, the application labels irreversible actions as such and clearly separates actions that are reversible from those that are not.

3.6.7 Use of Processing Modes

Processing modes are states where user actions have different results depending on the mode currently in effect. Both Motif and Windows use modes when defining how users interact with some elements of the user interface. For example, the mode assigned to a dialog window determines the extent to which users can interact with other windows while the dialog window is displayed (see section 4.1.2.2). The application is designed so that the same action has the same effect whenever it is executed by users. If an application-specific processing mode is required for a particular function, users are provided with a visual cue to indicate the mode currently in effect. For example, when users select a drawing tool from a toolbar, the tool button remains highlighted to indicate the mode in which users are operating.

3.6.8 Consistency in Performing Operations

Developers need to identify the set of operations (e.g., updating data records, building map overlays) that are supported within the application and then specify the sequence of actions that users follow to perform each operation. This sequence allows users to complete the operation rapidly and efficiently (i.e., minimizes the number of keystrokes executed and windows opened). For example, if an application requires users to enter location information for a contact by clicking the position on a map, the application records the position and uses the information; users do not have to type the same position information in a later part of its dialog with the application. If the application does not record the position selected, then users do not have to perform the action (i.e., click on a position). Users are able to execute the same sequence of actions whenever they perform the same operation in the application.

4.0 WINDOWS AND WINDOW ICONS

4.1 WINDOW MANAGEMENT

4.1.1 Window Components

A window, shown in figure 4-1, consists of (1) a window frame containing components that provide access to window management functions and (2) an application or client area within the frame in which the application displays information and interacts with users. The window frame includes a title bar, Window menu, window control buttons, and a resize border.

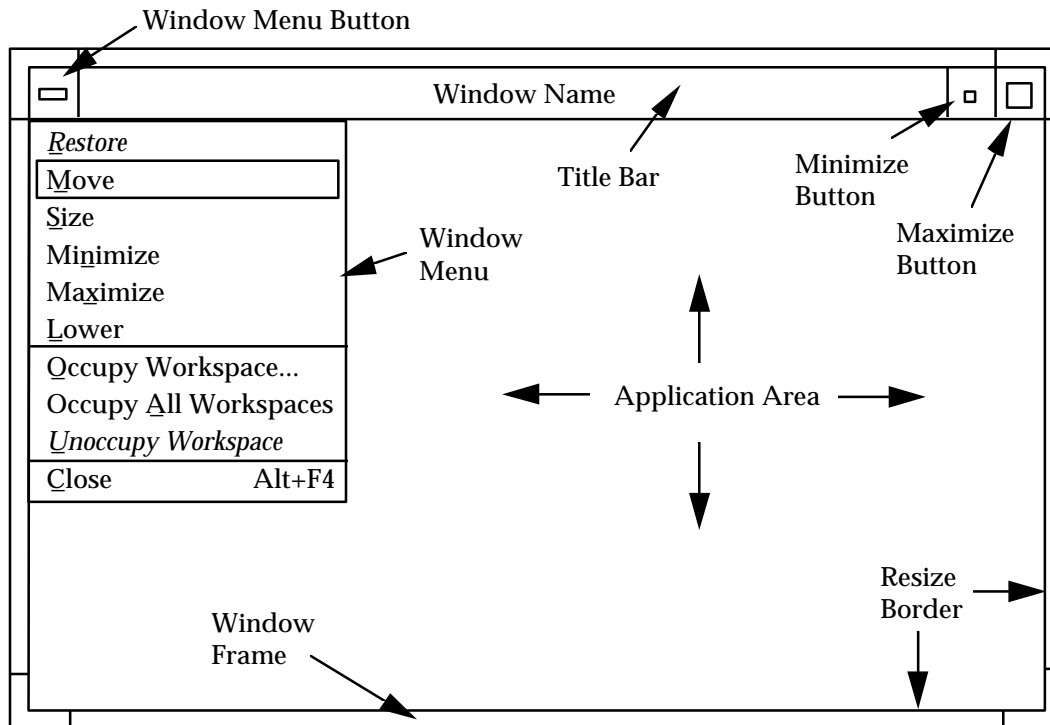


Figure 4-1. Standard window components in Motif.

4.1.1.1 Title Bar

The title bar extends across the top of the window, with the window name displayed in the middle of the bar. Clicking on the title bar raises the window to the front of the screen and gives it input focus. Pressing BSelect or BTransfer on the title bar and then dragging the pointer moves an outline of the window as the pointer moves. Releasing the button on the pointing device places the window at its new location.

4.1.1.2 Window Menu

If a window supports any window management functions, it has a Window menu with options for performing the functions. The Window menu button is displayed at the left edge of the title bar. Selecting this button displays a Window menu, as shown in figure 4-1. If a window management function is included in the menu, it executes the action and includes the mnemonic listed in appendix C. Motif and Windows recommend including an accelerator only for the Close option in a Window menu.¹¹ If the application includes accelerators for other menu options, it uses the ones listed in appendix C.

Motif Only: If any of the following window management functions is included in a Window menu, they are ordered: Restore, Move, Size, Minimize, Maximize, Lower, Occupy Workspace, Occupy All Workspaces, Unoccupy Workspace, and Close. Separators are included after the Lower option and before the Close option.

Windows Only: If any of the following window management functions is included in a Window menu, they are ordered: Restore, Move, Size, Minimize, Maximize, and Close. The Window

¹¹ Previous versions of Motif included accelerators for each option in the Window menu.

menu may also include Switch To and Next options, both of which follow Close. Separators are included before the Close option and after it if additional options are included in the menu.

The application can add window functions to the Window menu only under extraordinary circumstances. If functions are added, they are appended to the bottom of the menu, with a separator between Close and the added options.

Spring-loaded and posted menu selection methods (see section 5.1.2) are used to display the Window menu and activate options with the pointing device. Double clicking on the Window menu button closes the window. In an application window, <Alt><Space> (or <Shift><Escape> in Motif) selects the Window menu button, displays the Window menu, and moves the location cursor to the first available option in the menu.

Windows Only: In a document window, <Alt><Hyphen> selects the Window menu button, displays the Window menu, and moves the location cursor to the first available option in the menu.

The arrow keys move the location cursor between available options in the Window menu, and <Enter> (or <Return>, <Select>, or <Space> in Motif) activates an option and dismisses the menu. <Esc> (or <Cancel> in Motif) dismisses the menu without activating an option and returns the location cursor to its position before the menu was displayed.

4.1.1.3 Window Control Buttons

The Maximize button is displayed at the right edge of the title bar. The graphic in the Maximize button is a large square in Motif and an up arrow in Windows.

Motif Only: Activating the Maximize button in a normal-size window expands it to its maximum size. Activating this button in a maximized window restores the window to its size and location before being maximized.

Windows Only: Activating the Maximize button in a normal-size window expands it to its maximum size and replaces the button with a Restore button. Activating the Restore button in a maximized window restores the window to its size and location before being maximized.

The Minimize button is displayed in the title bar to the left of the Maximize button. The graphic in the Minimize button is a small square in Motif and a down arrow in Windows. Activating the Minimize button changes the window into a window icon.

If additional window management functions are defined by the application, they are mapped to buttons placed to the left of the Minimize button or to the right of the Window menu button.

4.1.1.4 Resize Border

No window components are placed outside the boundary formed by the resize borders. Pressing BSelect or BTransfer on the resize border and dragging the pointer moves an outline of the window as the pointer moves. The window is resized when the button on the pointing device is released.

Windows Only: All windows have frames except when they are maximized and fill the entire screen.

4.1.2 Behavior in Window Families

These specifications assume that an application uses one or more primary windows to provide access to data and associated operations for the top-level tasks in the application. The application displays secondary windows to present supplementary or supporting data or operations related to the primary window. These specifications define two types of secondary windows: a secondary task window (which is used for short-term interactions with data and controls related to the primary window) and a dialog window (which is used to present messages or conduct a brief dialog with the user).¹² This style guide refers to secondary task windows and dialog windows collectively as secondary windows, unless otherwise indicated.¹³

When users open a document or data file created by an application, the contents of the file are loaded into the application window that is opened. In Motif, the application can display only one file in the window at any given time. In Windows, the application can open a single document or multiple documents simultaneously. When the latter occurs, each document is displayed in a document window, all of which are placed within the application window. This style guide considers document windows to be a type of primary window and provides separate design specifications for them as needed to address their unique characteristics (see section 8.1.2.13).

4.1.2.1 Parent-Child Relationships

A window family consists of a main or parent window and one or more subordinate or child windows. The primary window is the parent for all other windows in the family. When a primary window is minimized, it and all of the secondary windows for which it is the parent are removed from the screen and replaced with a window icon. All processing in the primary window continues. When the window icon is opened, the window and all of its secondary windows that were displayed on the screen when the window was minimized re-appear. Each primary window in the application is minimized separately.

When a primary window is raised or lowered, it and all of the secondary windows for which it is the parent move with it. When a primary window is closed, it and all of the secondary windows for which it is the parent are removed from the screen, and all processing in the primary window stops. When the last primary window for the application is closed, the application is also closed.

Windows Only: A document window is a parent but also the child of the application window within which it is displayed.

When a secondary window is opened, it appears in front of its parent window, and the parent window remains displayed. When a secondary window is closed, it and any other windows that are its children are closed but its parent window is not affected. The preferred DII implementation is for dialog windows to have no children (except a help window, if one is available).

4.1.2.2 Modes of Interaction

The mode assigned to a child window determines the extent to which users can interact with other windows while the window is displayed. If a window is modeless, users can interact with other windows while it is displayed on the screen. If a window is modal, users are restricted from interacting with other application and/or system windows while it is displayed. A primary modal window does

¹² In Motif and Windows, secondary windows take the form of dialog windows so that the two terms are synonymous. The specifications presented here extend the basic definition of primary and secondary windows using the distinction made in the Bellcore style guide.

¹³ Another type of window in Motif is a menu window; it is described in section 5.3 on tear-off menus.

not allow interaction with any parent of the window, an application modal window does not allow interaction with any window created by the same application even if the application has multiple primary windows, and a system modal window does not allow interaction with any other window on the screen. Figure 4-2 illustrates the interaction restrictions for each type of modality.

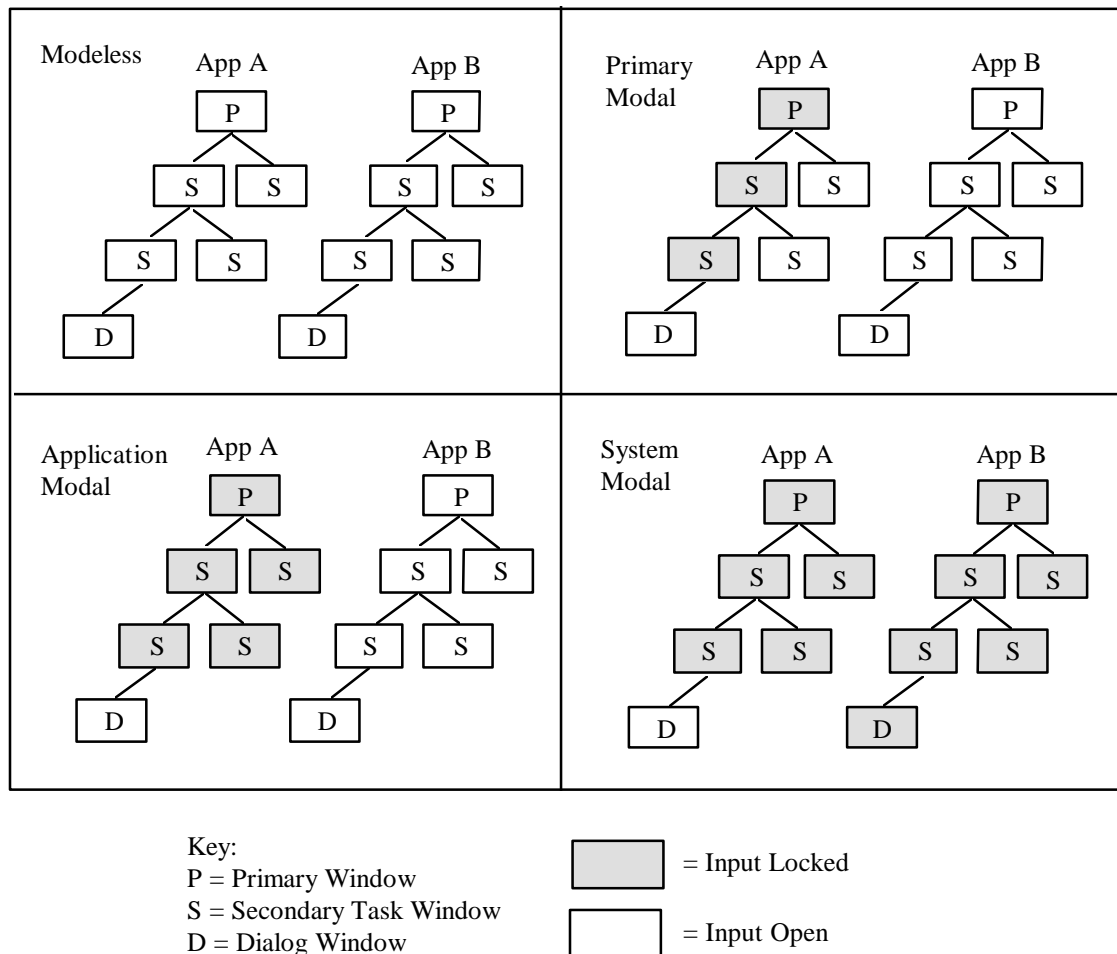


Figure 4-2. Interaction restrictions in modeless and modal windows.

Motif Only: Four types of modality are available: modeless, primary modal, application modal, and system modal. Child windows are modal only when the application cannot proceed without additional information (e.g., potential loss of data). Child windows are not system modal (i.e., block interaction with other applications) unless it is essential that users perform no other action on the desktop until they respond to the window. Similarly, child windows are not application modal unless it is essential that the state of the application remains unchanged until users respond to the window.¹⁴

Windows Only: Four types of modality are available: modeless, application modal, application semimodal, and system modal. Application semimodal windows are similar to application modal windows but permit limited interaction outside the window as a means of responding to the window. Child windows (e.g., message dialog windows) are modal (see section 8.2.1).

¹⁴ Previous versions of Motif required that certain types of dialog windows be modal.

4.1.3 Window Management Considerations

4.1.3.1 Window Size

The normal size of a window is large enough so that all of the objects in the window are visible when it is displayed on the screen. At a minimum, a window is wide enough to read the window title, and tall enough to read the contents of the title bar and menu bar.

When a window is maximized, more of the window content (i.e., objects and controls) is displayed but the size of the objects in the window does not change. Similarly, when a window is resized, the size of objects in the window and their relative position does not change. For example, if a text window is maximized, the size of the window increases, but the size of the text does not change. When a window is resized, the contents of the window remain visible so that users can view the effect of the change in size on the amount of information that can be seen.

A window can be resized larger only if more information will be visible in the window. The extent to which a window can be resized larger is limited so that any “restricted” areas of the screen (e.g., the classification bar) cannot be obscured. Likewise, the maximum size of a window is defined so that none of these areas is covered when an application window is maximized.

4.1.3.2 Window Arrangement

Windows can be arranged in either an overlapping or tiled placement. With overlapping placement, windows are stacked on top of one another; when a new window is opened, it is displayed in front of those already on the screen. A cascaded arrangement is a form of overlapping placement where windows are stacked one behind the other, keeping the title bar of each window visible. With tiled placement, windows are sized and positioned so that each one is completely visible at all times. The preferred implementation in the DII is overlapping placement.

4.1.3.3 Window Positioning

When a window is initially displayed, it is positioned on the screen so that it is completely visible. If a new window is related to other windows already displayed, it is positioned so that it does not obscure important information in the other windows. The new window is offset below and to the right of the information to which it relates so that the title of the window underneath remains visible (i.e., in a cascaded arrangement). If space is insufficient, the window is displayed to the left, below, or above the information. If a new window (e.g., a dialog window) is not related to other windows currently open, it is positioned in the center of the screen (or in the center of the application window in Windows). If a dialog window (or menu window in Motif) is already displayed but obscured by other windows, rechoosing the command that displayed the window raises it to the front of the window stack without affecting its position on the screen.

4.1.3.4 Processing in Minimized Windows

Users are informed when processing events occur in minimized windows. In Motif, a message window is displayed when a critical processing event (e.g., when processing stops, when an error occurs, when additional input is needed) occurs in a minimized window. In Windows, the icon for a minimized window flashes, with a message window displayed when the user restores the minimized window.

4.1.3.5 Moving Windows Between Workspaces (Motif Only)

Users may have several workspaces active on the desktop, with the application available in one or more workspaces. When the application opens a new window, it is displayed in the user's current workspace and only occupies that workspace. When the user moves application windows between workspaces, those windows related to a particular task move together. For example, the secondary windows for configuring a primary map window move with the primary window that is their parent; however, in an application with multiple independent primary windows (e.g., a set of word processing documents), moving one of the windows does not move the others.

4.2 WINDOW ICONS

4.2.1 Appearance

A window icon provides a visual representation of a minimized window or window family. When a window is minimized, the icon can be displayed directly on the workspace or in an icon box. A window icon consists of a graphic image and a label, as shown in figure 4-3.

Motif Only: The graphic image in a window icon is the same as that used for the application icon on the desktop.

The design of icon graphics is addressed in section 7.2.3.1 and appendix D. The label is located below the image and has the same title as the corresponding window.

Motif Only: When an icon does not have focus, its label is the same width as the icon image; the label is truncated as needed to fit. When an icon has focus, the location cursor is displayed on the icon and the full icon title displayed. A window icon includes a Window menu with the same options (except Size) as the Window menu of the corresponding window. Minimize can be included in the menu but is not available for selection.

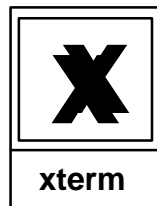


Figure 4-3. Example window icon in Motif.

4.2.2 Behavior

Placing the pointer on a window icon and double clicking BSelect restores a minimized window and its children. If the window had been maximized prior to being minimized, double clicking on the icon displays the window in its maximized size. Placing the pointer on an icon and dragging it with BSelect or BTransfer moves the icon.

Motif Only: Placing the pointer on a window icon and clicking BSelect displays the Window menu. Clicking BSelect anywhere outside the menu dismisses the menu. Navigating to an icon from the keyboard gives the icon focus and displays the Window menu. Selecting the Maximize option in the Window menu displays the window in its maximized size.

5.0 MENUS

5.1 PULL-DOWN MENUS

5.1.1 Appearance

A pull-down menu, shown in figure 5-1, consists of a menu title and a set of options displayed below the title from which users choose. The menu title is displayed in the menu bar at the top of the window and describes the category or type of options presented in the menu. The title is different from the other menu titles in the menu bar and does not appear as an option in the menu itself. The title is a single word whenever possible and does not contain numbers. The first letter of each word is capitalized, except for prepositions and articles, and includes a mnemonic (see section 5.5.5). If the title contains an acronym, it is capitalized. The title does not contain an ellipsis or a right-pointing arrow.

<u>E</u> dit	
<u>U</u> ndo	Ctrl+Z
<u>C</u> ut	Ctrl+X
<u>C</u> opy	Ctrl+C
<u>P</u> aste	Ctrl+V
C <u>l</u> ear	
<u>D</u> elete	

Figure 5-1. Example pull-down menu in Windows.

A pull-down menu can contain three types of menu options, shown in figure 5-2: (1) actions that are executed immediately, (2) routings that display a window or submenu, and (3) settings that define parameters or specify an application state. Routing options that display a window are followed by an ellipsis (i.e., "..."). Routing options that display a submenu are followed by a right-pointing arrow. Options that are settings can be action toggles (e.g., turn on/turn off) or state toggles (e.g., select a font size from the set available). Menu design, including the availability of mnemonics and accelerators, is addressed in section 5.5.

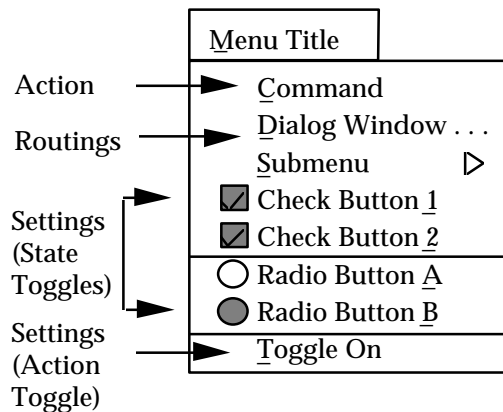


Figure 5-2. Examples of menu option types in Motif.

Submenus, shown in figure 5-3, are used to present groups of related actions hierarchically and shorten menus that are overly long. A submenu is positioned either to the right of the parent menu or below the parent menu if space to the right is limited. The first option in the submenu is aligned with the right-pointing arrow in the parent option for the submenu. The parent option is always shown as available. When this option is activated, the submenu is always displayed, even if all of the options in the submenu are unavailable. A submenu contains only the options in that menu and does not repeat the parent option as the first option in the submenu.

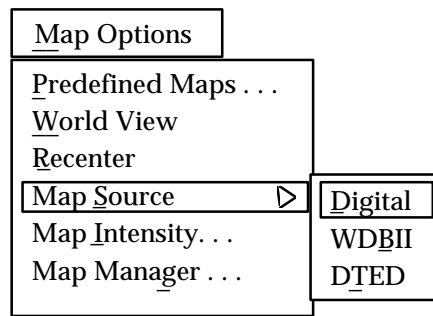


Figure 5-3. Example submenu in Motif.

5.1.2 Behavior

Spring-loaded and posted methods are used to display a pull-down menu and activate a menu option with the pointing device.¹⁵

In the spring-loaded method, users place the pointer on the menu title and press BSelect (or BMenu in Motif).¹⁶ The menu is activated (i.e., displayed), with the location cursor on the first available option in the menu. As users drag the pointer over the menu options, the location cursor tracks the pointer, moving to each option as the pointer is placed on it, and any submenus for which an option is the parent are displayed. The submenu is dismissed when the pointer moves off the parent option. Users drag the pointer to the option desired and release BSelect to activate the option and dismiss the menu. Users drag the pointer off the menu and release BSelect to dismiss the menu without activating an option.

In the posted method, users place the pointer on the menu title and click BSelect or (BMenu in Motif). The menu is displayed, with location cursor on the first available option in the menu. To display a submenu, users place the pointer on the option that is the parent for the submenu and click BSelect. To select an option, users place the pointer on the option desired and click BSelect. The location cursor moves to the option, the option is activated, and the menu is dismissed. Users move the pointer off the menu and click BSelect to dismiss the menu without activating an option.

<F10> (or <Shift><Menu> in Motif) moves the location cursor to the first (i.e., leftmost) available menu title in a menu bar; if none of the menu titles is available, then these keys do not move the location cursor to the menu bar from its position in the window. <Left> and <Right> move the location cursor between available menu titles in the menu bar, with the cursor wrapping between the last and first titles. <F10> (or <Shift><Menu> in Motif) exits the menu bar, and the location cursor returns to the object that had input focus before the menu was activated.

When the location cursor is on a menu title, <Down> displays the menu, with the location cursor appearing on the first available option.¹⁷ The arrow keys move the location cursor between available

¹⁵ This document does not call for the assignment of default options in pull-down or pop-up menus although Motif supports this feature as optional.

¹⁶ In previous versions of Motif, only BSelect was used to activate a pull-down menu. In CDE, applications can optionally support the use of either BSelect or BMenu.

¹⁷ The Motif Style Guide indicates that <Down> displays the pull-down menu while the Motif User's Guide indicates that <Return>, <Select>, or <Space> performs this action.

options, with wrapping from the bottom to top options in the menu. If the menu option with the location cursor is the parent for a submenu, <Right> displays the submenu and moves the location cursor to the first available option in it. <Up> and <Down> move the location cursor between options in the submenu, and <Left> dismisses the submenu and returns the location cursor to its parent option. <Enter> (or <Return>, <Select>, or <Space> in Motif) activates an option and dismisses the menu. <Esc> (or <Cancel> in Motif) dismisses the menu without activating an option and returns the location cursor to the object that previously had focus. Menu options can also be activated from the keyboard using mnemonics and accelerators (see section 3.4.2).

5.2 POP-UP MENUS

5.2.1 Appearance

A pop-up menu, shown in figure 5-4, is used to provide redundant access to frequently executed actions.

Motif Only: The menu includes a title describing the function performed by the menu or the object to which the menu relates. The title is displayed at the top of the menu and separated from the first menu option by a separator line.

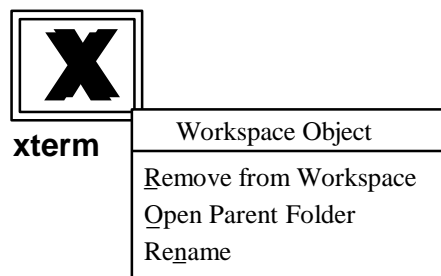


Figure 5-4. Example pop-up menu in Motif.

If a pop-up menu contains options taken from different menus in the window, options (e.g., Open, Save) pertaining to the object to which the menu relates are presented first, followed by edit options (e.g., Undo, Cut, Copy) and then by application-specific options. A separator delimits each group of options. A pop-up menu does not include any submenus.

Motif Only: If a pop-up menu includes any of the following options, they are ordered: Properties, Undo, Primary Move, Primary Copy, Primary Link, Cut, Copy, Copy Link, Paste, Paste Link, Clear, Delete, Select All, Deselect All, Select Pasted, Reselect, and Promote. If the menu relates to a selectable object, it also includes Move To, Copy To, Put in Workspace, Delete, and Help options.

When the pointing device is used to display a pop-up menu, the menu contents relate to the object under the pointer. When the keyboard is used, the menu contents relate to the element with the location cursor. In both cases, the pop-up menu is displayed near the element with which it is associated.

5.2.2 Behavior

Users display a pop-up menu by pressing or clicking BMenu and navigate within the menu and activate an option with either BSelect or BMenu. Releasing BMenu with the pointer in the area that displayed the menu dismisses it without activating an option.

Motif Only: Clicking BSelect or BMenu outside a pop-up menu also dismisses it. When the pop-up menu relates to a selection, the action executed by the option chosen applies to all of the elements in the selection. When a pop-up menu relates to an unselected object, the action executed applies only to that object and not to any other object(s) that might be selected.

Windows Only: If there is no current selection, clicking BMenu on an object both selects it and displays its pop-up menu. Clicking BSelect outside a pop-up menu removes the menu and selects the object under the pointer if one is there; clicking BMenu outside the menu only removes the menu.

In Motif, <Shift><F10> (or <Menu>) displays a pop-up menu if one is available; Windows does not support keyboard activation of a pop-up menu. In both Motif and Windows, the arrow keys move the location cursor between options in a pop-up menu. <Enter> (and <Return>, <Select>, or <Space> in Motif) activates an option in the menu and dismisses it. <Esc> (and <Cancel>, <Menu>, or <Shift><F10> in Motif) dismisses a pop-up menu without activating an option, and the location cursor returns to the object that had input focus before the menu was displayed.

5.3 TEAR-OFF MENUS (Motif Only)

5.3.1 Appearance

A pull-down menu, submenu, or pop-up menu provides a tear-off capability if users need to select repeatedly from the menu. A menu that can be torn off contains a tear-off button with a dashed-line graphic that is the first menu option below the menu title, as shown in the top part of figure 5-5. The tear-off button behaves in the same manner as other available options in the menu. When a tear-off menu is displayed, the location cursor appears on the second available option if the first option is the tear-off button.

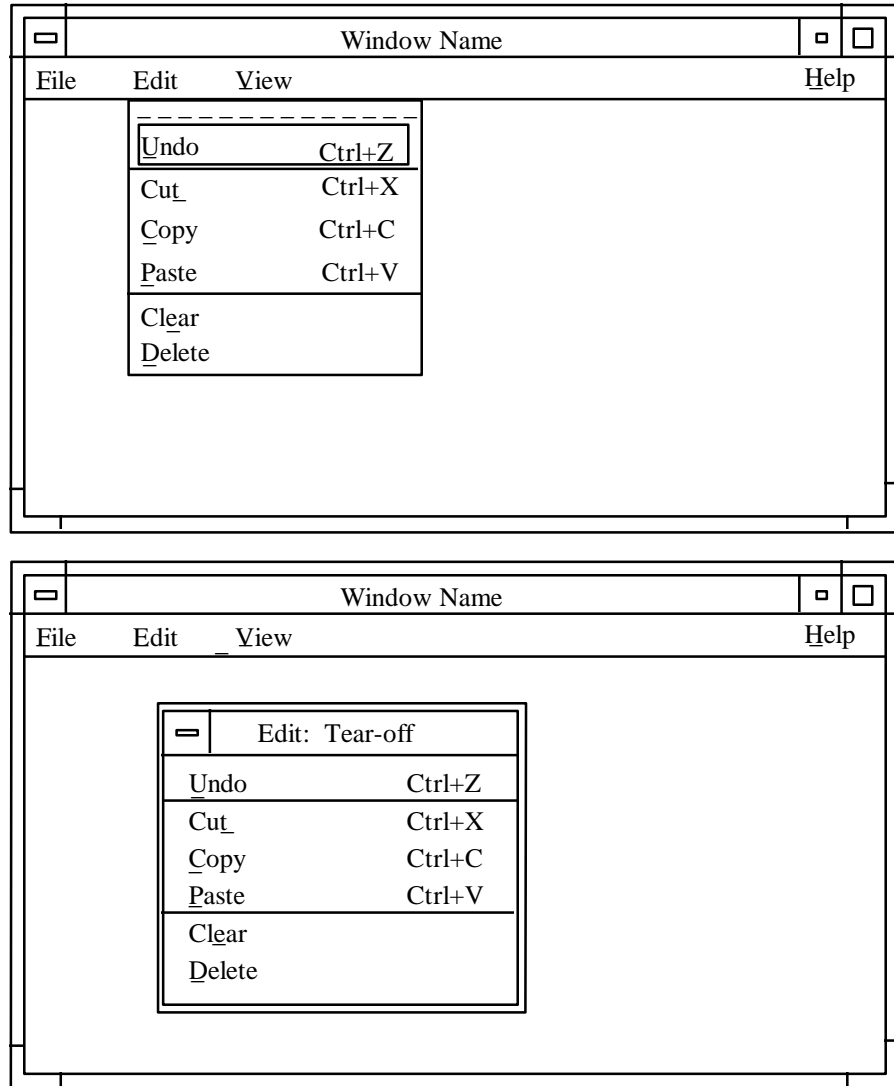


Figure 5-5. Example tear-off menu and menu window in Motif.

When users activate the tear-off button, the options in the menu are displayed in a menu window, as shown in the bottom part of figure 5-5. The title of the window is the title of the associated menu. The contents of the menu window are the same as the original menu, with the options presented in the same order in both cases, but the tear-off button is not displayed in the window. The options in the menu window are dimmed when unavailable and behave in the same manner as the options in the original menu (e.g., the wording of an action toggle changes when selected).

5.3.2 Behavior

Placing the pointer on the tear-off button and clicking BSelect dismisses the menu and displays a menu window at the menu location; dragging the tear-off button with BTransfer displays a menu window and moves it to a new location. In both cases, the window receives focus, with the location cursor on the first available menu option. Clicking BSelect on an option in a menu window moves the location cursor to the option and activates it; the window remains displayed so that the menu is available for additional selections.

The arrow keys move the location cursor to the tear-off button in a menu. <Return>, <Enter>, <Select>, or <Space> dismisses the menu, displays a menu window at the menu location, and assigns input focus to the window, with the location cursor on the first available menu option. The arrow keys move the location cursor between available options in the menu window; <Return>, <Enter>, <Select>, or <Space> activates an option.

A menu window can be moved but not minimized or maximized. A tear-off menu can be displayed while the menu window is open, and options can be activated from either the menu or the window. If the tear-off option in the menu is activated, the original menu window is dismissed and replaced with a new version of the window. The menu window remains displayed until dismissed, either by selecting Close in the Window menu or pressing <Esc> or <Cancel>.

5.4 OPTION MENUS (Motif Only)

5.4.1 Appearance

An option menu, such as the one shown in figure 5-6, consists of a title and an option button which, when activated, displays the menu. The title is placed to the left of the option button (or above it if space is limited) and followed by a colon. The label in the button is the most recent option selected from the menu and includes a bar graphic. The option button is large enough to display the longest menu option and the bar graphic; the graphic does not obscure any text in the button.

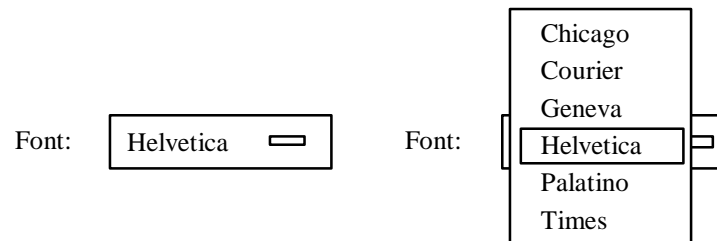


Figure 5-6 Example option menu in Motif.

5.4.2 Behavior

Spring-loaded and posted methods described in section 5.1.2 are used to display an option menu, navigate in the menu, and select a menu option using BSelect or BMenu.¹⁸ When an option is selected, it is displayed as the label in the option button and the menu is dismissed.

When focus is on the option button, <Space> or <Select> displays an option menu, with the location cursor on the previously selected option in the menu. The arrow keys move the location cursor between available options in the menu. <Return>, <Enter>, <Select>, or <Space> selects an option, dismisses the menu, and displays the option in the option button. <Esc> or <Cancel> dismisses an option menu, and the location cursor returns to the object that had input focus before the menu was displayed.

5.5 MENU DESIGN GUIDELINES

5.5.1 Format of Menu Options

¹⁸ In previous versions of Motif, only BSelect was available to display an option menu and select an option.

Menu options are presented in mixed case, with the first letter of each word capitalized, except for prepositions and articles. If the option contains an acronym, it is capitalized. The menu is wide enough to accommodate the widest option and an accelerator (if one is available). The accelerator includes a plus sign to indicate the combination of keys (e.g., Shift+Ins) that must be pressed at the same time. The end of the menu option is separated from the start of the accelerator by at least three character widths. Menu options and accelerators are left-justified and appear on a single line; long menu options are accommodated by making the menu wider rather than making the item take two (or more) lines.

5.5.2 Wording of Menu Options

Menu options are phrased to reflect the action that is executed by the option (i.e., phrased as a command to the application rather than as a question to the user), and worded in the vocabulary of users rather than that of application developers. Appendix C lists the vocabulary that the application is to use when the actions described are included in menu options.

Menu options that are action toggles are worded to reflect the action that is implemented when the option is activated. The wording is semantically congruent with natural usage; for example, if one toggle is worded “Move Object Up,” the other toggle is “Move Object Down,” not “Move Object Back.” When users activate one of these options (e.g., turn on), the wording of the option in the menu changes (e.g., turn off) to reflect the action that will be implemented when the action is executed again. Only one of the action toggles appears in the menu at any time. If an Undo option is provided, the wording changes dynamically to reflect the action that can be undone. For example, if the most recently executed action is cut, the option is worded “Undo Cut.”

Options that are state toggles are worded to describe the state (e.g., a list of font names) being set.

Motif Only: Options that are state toggles include a radio button (for an exclusive setting) or a check button (for a nonexclusive setting) to the left of the option. When users activate one of these options, the select state of the check button or radio button changes but the option wording remains the same. The button appears as either selected or unselected and is not removed from the menu when deselected.

Windows Only: Options that are state toggles do not include a radio or check button to the left of the option. When users activate an option in a set of nonexclusive settings, a check mark is placed to the left of the option. When users activate an option in a set of exclusive settings, a dot is placed to the left of the option. When the state of a setting is indeterminate (e.g., selected text includes both normal and bold font sizes), the graphic(s) is removed from all groups for which the setting is indeterminate.

If radio buttons are used in a menu, they are grouped together and delimited from other menu options by a separator.

The wording of each menu option is consistent in grammatical style and matched with the corresponding menu title. For example, additions to an Edit menu containing actions such as Cut, Copy, Paste, and Undo are other verbs describing actions, rather than nouns describing objects or entities. Verbs are used as the first word in options in pull-down and pop-up menus because this format provides consistency and makes the menu options easier to read and understand. If desired, supplementary information about menu options can be provided in the message bar at the bottom of the window containing the menu; as the pointer is placed on an option, a more complete prose description of the action executed by the option is displayed.

5.5.3 Grouping into Menus and Submenus

A pull-down or pop-up menu contains no less than two or more than 15 options, and an option menu contains no more than 10-12 options. Menus with more than four options are divided into groups based on function (as shown in figure 5-7), with each group containing no more than four options (unless more are logical) and delimited by a separator line between groups. If the options cannot be organized into logical groups, they are ordered according to frequency of usage, with the most frequently executed at the top of the menu and least frequently executed at the bottom. If an organization based on logical groups or frequency of use is inappropriate, then the options are ordered alphabetically or in numerical order. Destructive options such as Delete and Clear appear at the bottom of the menu. Menu options that perform opposing actions (e.g., Save and Delete) are not placed adjacent to each other in order to reduce the likelihood of accidental selection of an incorrect action. If similar options are included in different menus, the options are ordered in a consistent manner in each menu.

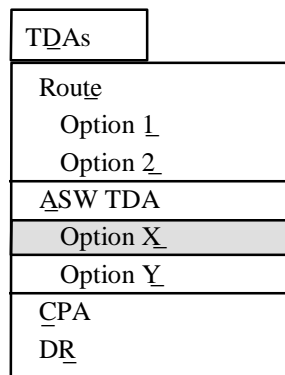


Figure 5-7. Example of logical ordering of menu options in Windows.

Submenus are limited to two levels below the parent menu option and organized as shown in figure 5-8. If the number of submenus will exceed two levels, new menus are created or a dialog window is used. Submenus contain at least three options; if there are only two options, they are placed in the parent menu rather than creating a separate submenu and putting them there.

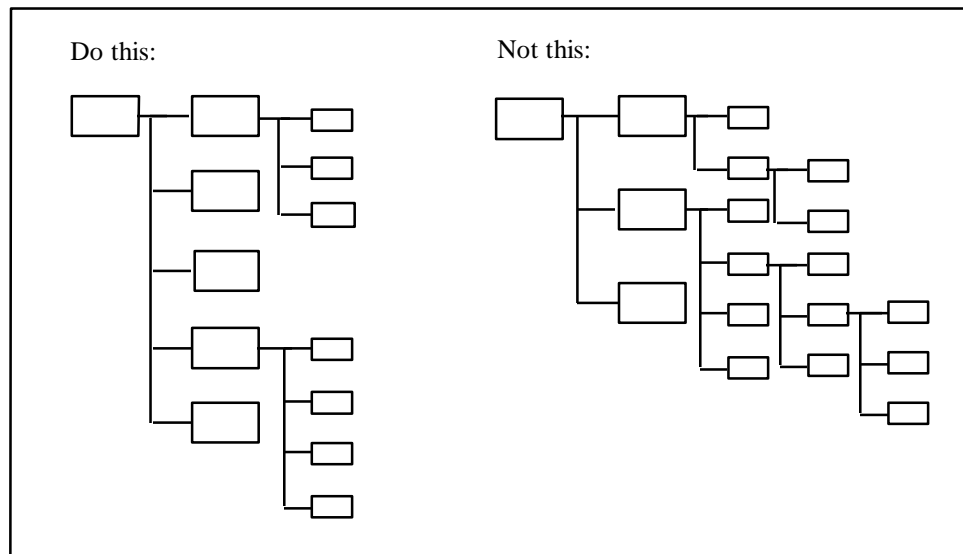


Figure 5-8. Organization of options in submenus.

5.5.4 Availability of Menu Options

If a menu option is never available to a user (e.g., system administrator commands), it is not included in a menu. If an option is only temporarily unavailable, it is displayed in the menu but dimmed (as in figure 5-9) to indicate that it cannot be selected. The wording of options may change (e.g., when an option is an action toggle). However, options are not added to or deleted from a menu to indicate their availability within a particular part of an application.¹⁹

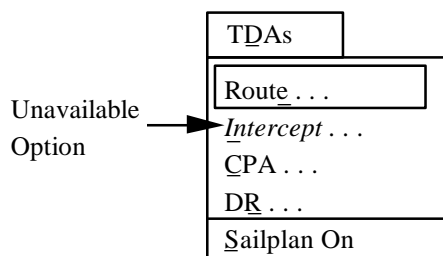


Figure 5-9. Example of an unavailable menu option in Motif.

5.5.5 Mnemonics and Accelerators in Menus

The application provides mnemonics in all pull-down menu titles and options. The mnemonics in appendix C are used whenever one of the actions listed therein is implemented in the application.

Motif Only: Mnemonics are also available in pop-up menus and tear-off menus.

Windows Only: Mnemonics are not available in pop-up menus.

The character assigned as the mnemonic is included in the text label of the menu title or option and underlined, as shown in figure 5-10. Whenever possible, the mnemonic is the first character of the label; a distinctive consonant in the label is preferred over vowels. If the mnemonic is not the first character, it is the last character of the label or the first character of the second word (if there is more than one word) or a sequential number that is assigned to the label. If a mnemonic does not appear in the label, it is placed in parentheses following the text. Mnemonics are not case sensitive; users can type the character in either upper or lower case.

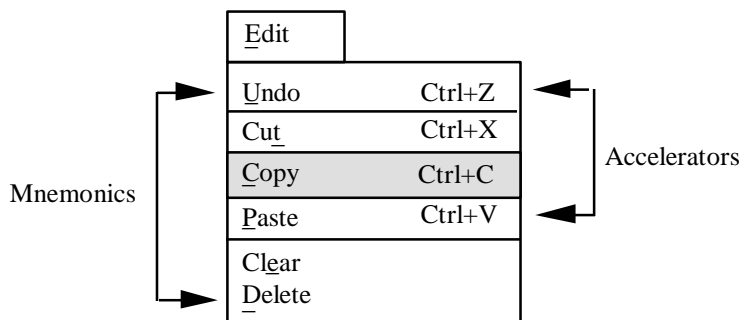


Figure 5-10. Example mnemonics and accelerators in Windows.

¹⁹ The inclusion of unavailable options in a menu is consistent with Motif and Windows but not with MIL-STD-1472D.

The characters selected as mnemonics in each menu bar and each menu are unique, but the same character can be used as a mnemonic in different menus. A menu title or option has the same mnemonic whenever it appears in the application. Mnemonics are selected to produce minimal interference with the mnemonics used in other menus. For example, the same character is not assigned as the mnemonic for options performing opposite or contradictory actions in different menus (e.g., C used as the mnemonic for Continue in one menu and Close in another).

The application provides accelerators for frequently executed actions in pull-down menus. The same key combination is used for a given accelerator throughout the application.

Motif Only: Accelerators are available in pop-up menus only if the menu contains the same options as a pull-down menu and uses the same accelerators as the corresponding options in the pull-down menu.

Windows Only: Accelerators are not available in pop-up menus.

When an accelerator is activated, the associated menu is displayed briefly and its action is applied in the window that has focus.

The accelerators in appendix C are used whenever one of the actions listed therein is implemented in the application. If new accelerators are created, they have the form “modifier+character,” where the modifier is <Alt>, <Ctrl>, <Shift>, or a combination of these keys, and the character is an alphanumeric or special key on the keyboard. Mnemonics and accelerators are selected to be coordinate with each other. For example, if S is the mnemonic for Save, Ctrl+S is used as the accelerator.

6.0 CONTROLS

6.1 PUSH BUTTONS

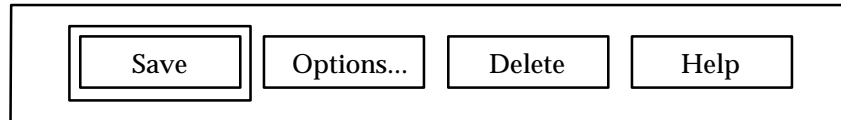
6.1.1 Appearance

A push button is used to initiate an action. The button contains a text or graphic label, as shown in figure 6-1, indicating the action executed when the button is activated. The label is centered, and enough space is provided between it and the rectangle surrounding it so as not to restrict the legibility or visibility of the text or graphic in the push button. The push button designated as the default has an extra border around it.

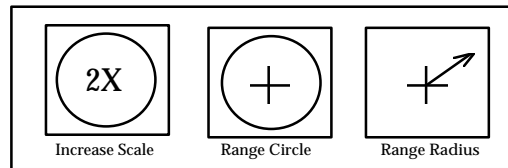
Motif Only: Push buttons have the default dimensions defined by Motif, except for margin width which is 8 pixels (per Kobara).²⁰

Windows Only: Push button height is 22 pixels, when the default font is used (see section 9.1.1).

²⁰ Resource settings that implement visual design recommendations in Kobara are provided in appendix D.



Push Buttons with TextLabels



Push Buttons with Graphic Labels

Figure 6-1. Example push buttons in Motif.

All of the push buttons in a group are the same size. If the buttons have text labels, the buttons are wide enough to display the longest button label. If the buttons have graphic labels, the buttons are large enough to display the largest graphic. Exceptions to these size guidelines may occur in order to accommodate a button with a label or graphic that is significantly longer or larger than the others in the group, especially when space in a window is limited. The first letter of each word in a text label is capitalized, except for prepositions and articles. The label includes an ellipsis if activating the button results in another window (other than a help window or a message window to confirm the action executed by the push button) being displayed.

The vocabulary in appendix C is used to construct push button labels whenever the application performs the actions listed. New vocabulary can be created as needed to describe actions not in the appendix. New vocabulary is not created for actions already defined, and existing vocabulary is not used to describe new actions. If new vocabulary is created, it is a verb and stated in active voice; it describes the results of pressing the button and reflects the action that will be taken by the application rather than the user. The names of actions are congruent (e.g., Save/Delete, On/Off, In/Out), with the same vocabulary used to describe an action throughout the application. Terms such as “All” are used in labels (e.g., Select All, Delete All) only when there is no ambiguity as to the object or data element to which “All” refers. If a label may have multiple referents within a window, the name of the object or data element is used instead of “All.”

Action icons (i.e., push buttons with graphic labels) are designed so users can easily identify the function performed when the icon is selected. Each graphic is unambiguous and easily distinguished from the action icons with which it is displayed. The graphics for action icons that represent opposite functions (e.g., Save, Delete) are designed to mirror each other. Action icons do not contain an ellipsis, even if activating the push button with the graphic displays another window.²¹ Section 8.1.2.8 describes the design of action icons in toolbars.

6.1.2 Behavior

Placing the pointer on a push button and clicking BSelect activates a push button. When the location cursor is on a push button, <Space> (or <Select> in Motif) activates it from the keyboard. When a push button is activated, it highlights (i.e., changes color) and the action represented by the push button is executed. Releasing BSelect outside the push button does not activate the push button. <Enter> (or <Return> in Motif) activates the default push button in a window.

²¹ A set of action icons for frequently executed actions will be included in a future version of this document.

6.2 RADIO BUTTONS

6.2.1 Appearance

Radio buttons are used in groups to select one option from a set of mutually exclusive options. A radio button consists of a circular indicator, followed by a label describing the option represented by the button, as shown in figure 6-2.²²

Motif Only: Radio buttons have the default dimensions defined by Motif (per Kobara).

The radio button label defines the state being set by the user. The first letter of each word in the label is capitalized, except for prepositions and articles. If a radio button cannot be selected, its label is grayed out to indicate its unavailability.

Motif Only: When a group of radio buttons is displayed, one of the buttons is selected. If users need to select none of the options in the group, a radio button labeled None is provided as an option, rather than allowing users to deselect all of the buttons in the group.

Windows Only: Radio buttons can be used to represent a fixed or heterogeneous set of attributes for selection. In the former case, the radio button corresponding to the current attribute is selected when the buttons are displayed. In the latter case, all of the buttons in the group are deselected when displayed, and choosing any button applies the option to the entire selection.

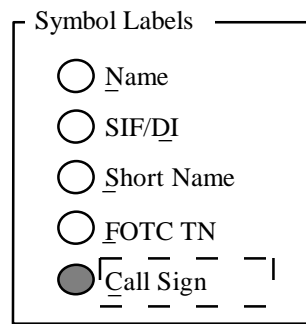


Figure 6-2. Example radio buttons in Windows.

6.2.2 Behavior

Placing the pointer on a radio button or its label and clicking BSelect selects the button.

Motif Only: When focus is on a radio button, pressing <Space> or <Select> selects the button.

Windows Only: Moving the location cursor to a radio button selects it from the keyboard.

When a radio button is selected, it highlights and any previously selected button in the group is deselected. Only the select state of the option changes; selecting a radio button does not initiate an

²² In previous versions of Motif, radio buttons had diamond-shaped, rather than circular, indicators.

action or display a dialog window. If the radio button is in a window with a default action, pressing <Enter> (or <Return> in Motif) in a radio button selects the button and executes the action.

6.3 CHECK BUTTONS

6.3.1 Appearance

Check buttons are used singly or in groups to indicate a nonexclusive setting; selecting a check button toggles to the setting or state indicated by the label. A check button consists of a square-shaped indicator, followed by a label describing the option represented by the button, as shown in figure 6-3.

Motif Only: Check buttons have the default dimensions defined by Motif (per Kobara).

The check button label defines the state being set by the user. The first letter of each word in the label is capitalized, except for prepositions and articles. If a check button is unavailable for selection, its label is grayed out to indicate its unavailability. A check button (rather than two radio buttons) is used if an option can only be set to on or off.

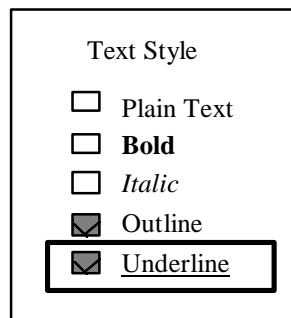


Figure 6-3. Example check buttons in Motif.

Windows Only: Check buttons that are used to set the properties of a selection can have three states: on, off, or indeterminate. A check button is indeterminate when the selection is heterogeneous; for example, a font style check button is indeterminate when the selected text contains both normal and bold styles. When the state of the check button is indeterminate, the indicator is filled with a gray pattern. Clicking on the check button cycles through the three states, with the appearance of the indicator reflecting the current state.

6.3.2 Behavior

Placing the pointer on a check button or its label and clicking BSelect selects the button. When focus is on a check button, <Space> (or <Select> in Motif) selects the button from the keyboard. When a check button is selected, the indicator highlights and contains a check mark²³ in Motif or contains an X in Windows; any previously selected check button in the group remains selected. When users select a check button, only the select state of the option changes; selecting a check button does not initiate an action or display a dialog window. If the check button is in a window with a default action, pressing <Enter> (or <Return> in Motif) in a check button selects/deselects the button and executes the action.

6.4 LABELS

²³ Previous versions of Motif indicated the select state of a check button only with highlighting.

A label is used to display static text and graphics in a window. Static text (e.g., titles, headings, and directions) is presented in a label and not in a text field. A label is not selectable, and it is not traversable from the keyboard. The first letter of each word in the label is capitalized, except for prepositions and articles.

Motif Only: Labels have the default dimensions defined by Motif, except for margin height which is 6 pixels so that labels align properly with other controls (per Kobara).

6.5 TEXT FIELDS

6.5.1 Appearance

A text field, shown in figure 6-4, is used to enter and edit text. A text field includes a label describing what is to be entered in the field. The label is placed either to the left or above the field and has the same background color as the window in which it appears. The label is followed by a colon, and the space between the colon and the text field is empty (i.e., does not include delimiters or underscores). The label is grayed out if the text field is unavailable for text entry.

Motif Only: Text fields have the default dimensions defined by Motif, except for margin height which is 2 pixels in order to be consistent with the appearance of other controls (per Kobara).

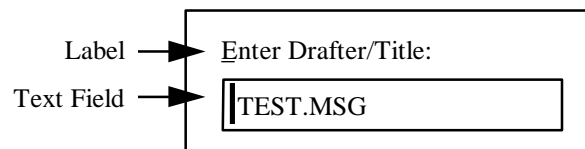


Figure 6-4. Example text field and label in Windows.

Cues regarding the format of the text to be entered in the text field are presented in the message bar of the window if one is present; otherwise, the label provides these cues. If a unit of measurement (e.g., feet, miles) is always associated with a field, it is displayed as part of the label, as shown in figure 6-5, and does not have to be entered by users. Cues regarding whether text entry is mandatory or optional are presented in the message bar if one is present; otherwise, these cues are provided in the label or color coded in the text field itself. The label is worded to be clearly different from the labels used in other text fields in the window. If the labels in a group of text fields are highly redundant (e.g., ship name, ship UIC, ship homeport), the common word is removed from the label (e.g., Name, UIC, Homeport) and placed instead in the heading (e.g., Ship Information) that describes the group of fields.

The diagram shows a window with four text fields. Each field has a label to its left and a format cue to its right. The labels are 'Date:', 'Distance:', 'Frequency:', and 'SSN:'. The format cues are '(YYMMDD)', 'Miles', 'MHz', and three empty boxes respectively. The text fields are represented by rectangular boxes.

Figure 6-5. Example text field labels providing format cues.

The text entry area is long enough for users to enter the information required. If the information is a fixed length, then the field is the same length as the information. If the information varies in length, the field is at least as long as the longest information. When window space is limited, the text field includes

scroll bars if the information is longer than the field or extends beyond a single line. Strings of characters (that are not a word) longer than five to seven characters are entered in smaller chunks. The field format is meaningful to users (e.g., year, month, and day in a date; the three parts of a Social Security number) and consistent with their expectations. Routine or default data, data already known by the application, or data that can be computed by the application is automatically entered in a field whenever possible. For example, if fields are provided for the start date, end date, and duration of a mission, users enter only two of the values and the application calculates the third.

These specifications call for editable text to be displayed in a text field and noneditable text to be displayed in a label. However, in some cases, the application may need to present application-generated data that is not editable in a text field. If dynamic noneditable text is displayed in a text field, the field has a different appearance (e.g., background color) than an editable text field. When the pointer is on a noneditable field, its shape does not change to an I-beam. A noneditable text field is not selectable; i.e., clicking on it does not change its appearance or display a text cursor in the field.

6.5.2 Behavior

6.5.2.1 Supporting Text Entry and Manipulation

Section 2.2.3 describes text entry modes and actions, and section 3.2.2.4 addresses text cursor behavior during navigation.

When variable-length information is entered in a text field, the information is automatically justified or truncated; users do not have to enter leading characters to fill the space available (see figure 6-6). Text characters are displayed as typed by users (with the same capitalization and punctuation) unless a special format is required (e.g., composing a message). When the application presents stored text in a text field, it is displayed in a consistent format so users can recognize it as such, and any text editing by users is converted into this format.

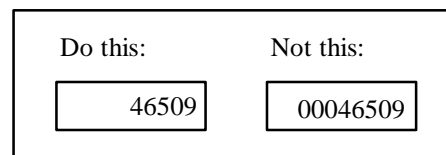


Figure 6-6. Example of automatic justification during text entry.

Users can enter numeric data from either the keyboard or the numeric keypad. The amount of data that has to be entered in a text field is minimized; for example, users are able to enter an abbreviation rather than an entire word and do not have to enter the unit of measurement associated with a number value. The application performs automatic entry of data into a text field (e.g., prefills a date/time group field with the current date and time, or a lat/long field with the current position of the user's ship). In addition, if the application supports position hooking (i.e., allow users to click on a geographic location and have the coordinates of the position entered into a lat/long field), it provides an indicator in the windows where this feature is available (e.g., in the label for the field, with a special symbol next to the field).

Users are not required to enter data in a mandatory field before moving to another field. Users can accept the default value in a text field by tabbing to next field in a window; tabbing does not affect the default. If users modify the default but do not save the change, the change does not affect the default when the window containing the field is displayed again.

Autotabbing is available in text fields only when data such as date, time, latitude, and longitude are broken into smaller groups of characters, with each group entered in a separate text field (e.g. in the SSN field in figure 6-5). In this case, autotabbing can be used since users consider the characters to be a single data value and expect to enter the data without the need to tab between the fields; while separate text fields are intended to improve readability and minimize the opportunity for error, they should not interfere with efficient data entry by users.

6.5.2.2 Error Checking and Correction

The application performs error checking on the data entered and provides feedback when errors are detected. Users can fix errors by editing individual characters in the field, rather than having to erase and retype the entire field.

When users enter data in fields that are unrelated to one another, they receive feedback about an error (e.g., data in an incorrect format or outside range requirements) when they tab out of the text field but they are not prevented from leaving the field. Error feedback is provided in the message bar of the window if one is present; otherwise, feedback is provided by changing the appearance of the field with the error and/or in a separate message window. When users enter data in fields that are interdependent (i.e., the data in one field are correct or in error based on data entered in other fields), they receive feedback (e.g., in the message bar or a message window) concerning the errors when they attempt to accept or save the data.

Users can save the data they have entered at any time by executing an explicit action (e.g., activating a Save, Apply, or OK push button). The application informs them if any errors are present and does not accept (i.e., save) the data until all errors are corrected. The application provides feedback (e.g., a confirmation message) to indicate the data have been saved. When users enter data, all are saved, regardless of text cursor position in the window.

6.6 LIST BOXES

6.6.1 Appearance

A list box such as the one shown in figure 6-7 is used to select one or more items from a set of items. The items in the list are displayed vertically, with one item per line. A vertical scroll bar appears to the right of the items in the list when the number of items exceeds the space available. If the list box includes a title, it is a label describing the purpose or contents of the list; the label appears above the list and is not followed by a colon.

Motif Only: Lists have the default dimensions defined by Motif, except for margin height and width, both of which are 2 pixels in order to provide space between the list borders and contents (per Kobara). Kobara also recommends that a vertical scroll bar be displayed even when the list is short enough to fit in the display area.

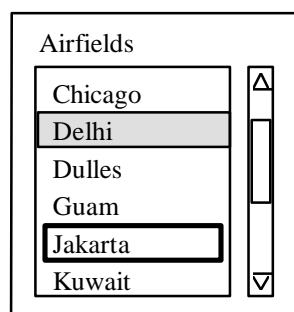


Figure 6-7. Example list box in Motif.

Windows Only: If a list is inactive (i.e., unavailable), its title is dimmed. If an item in a list is unavailable for selection, it is omitted from the list (rather than displayed as unavailable). If scrolling is not possible (e.g., the list is inactive, all of the items are visible), the stepper arrows are dimmed, the slider is removed, and the color of the trough region is changed to that of the window background.

The size of a list box depends on the amount of space available in the window in which it is displayed. A list is large enough to display six to eight items at a time, or all of the items if there are fewer than six. A list is wide enough so that users can read all of the items without scrolling horizontally; if items differ significantly in length, the list is wide enough to display the items of average length and includes a horizontal scroll bar to allow users to read the longer items.

Motif Only: A horizontal scroll bar appears automatically next to the list, rather than increasing the display area, when the width of the list grows beyond that area (per Kobara).

If the items in a list represent possible attribute values for a selection, the current value is selected (and highlighted) when the list is first displayed. If the selection is heterogeneous, none of the items is selected when the list is displayed.

The items in a list are presented in sequential order based on the nature of the items and the sequence in which users expect the items to occur (e.g., chronological, alphabetical, sequential, functional, by importance). For example, a list of port names is ordered alphabetically, and a list of messages by precedence, date-time group, or a combination of the two (e.g., date-time group within precedence). Selecting an item in a list does not affect the order of the items. When items are added to a list, they appear in their correct position within the list (e.g., in numerical or alphabetical order) rather than at the end of the list. Whenever the content of a list is updated through an automatic process, the list does not scroll automatically to the item that was added; the list scrolls only in response to a user action (e.g., using a scroll bar).

6.6.1.1 Drop-Down List Boxes (Windows Only)

A drop-down list is used instead of a regular list box when the space available in a window is limited. A drop-down list consists of a noneditable text area showing the currently selected item, a down-pointing arrow button, and a list of items that is displayed when the arrow button is depressed. The same guidelines concerning height and width apply to an open drop-down list as to a regular list. The arrow button abuts the text area in drop-down list box (in contrast to a drop-down combo box where there is a small gap between the field and the arrow button). When users select an item in the list, it is displayed in the field and the list closes.

6.6.1.2 Multiple Selection List Boxes (Windows (Only))

Windows distinguishes between list boxes supporting single selection and those supporting multiple selection. A multiple selection list is used when users need to select multiple discontinuous items. This type of list includes a check button preceding each item (as a visual cue that multiple items can be selected). When users select an item in the list, an X appears in the check button for that item.

6.6.2 Behavior

6.6.2.1 Navigation and Selection

Placing the pointer on a list item and clicking BSelect moves the location cursor to the item and selects it. If other pointing device selection methods are available in the list, they are performed as

defined in table 3-1. If the window containing the list has a default action, double clicking on an item selects the item and executes the action.

<Up> and <Down> move the location cursor to the previous and next items in a list; <Left> and <Right> scroll the list one character to the left and right. <PageUp> and <PageDown> move the location cursor to the item one page up and down in the list; <Ctrl><PageUp> and <Ctrl><PageDown> scroll the list one page to the left and right. <Home> and <End> scroll to the leftmost and rightmost edge of the list; <Ctrl><Home> and <Ctrl><End> move the location cursor to the first and last items in the list. Selection is performed from the keyboard as defined in table 3-2 for Motif applications and in table 3-3 for Windows applications.

6.6.2.2 Speed and Incremental Search in Lists²⁴

A speed search capability is available in lists containing more than 10 items. When the list has focus and users type a character, the list scrolls to the first instance of an item that begins with that letter, and the location cursor moves to that item. In Windows, the item is also selected. When users type the character again, the list scrolls (as needed) and the location cursor moves to the next item that starts with the character.

An incremental search capability is available in lists containing more than 50 items. In this case, a text field is provided with the list, as shown in figure 6-8. Users can type the first few letters of the item desired in the text field or enter wild card characters to search for specific text patterns, as described in section 9.1.5. When they press <Enter> (or <Return> in Motif), the list scrolls to the first occurrence of an item that matches the letters. Users can scroll through this part of the list to locate the item desired or type additional characters to narrow the search further.



Figure 6-8. Example list and text field used in an incremental search in Motif.

In both speed search and incremental search, if the character(s) typed do not match any of the items in the list, the location cursor does not move, and users receive feedback (e.g., an auditory signal, a message in the message bar, or a message window) to indicate that no match was found. Both search and incremental search are not case-sensitive; if the search has to be case-sensitive, then this information is provided to users (e.g., in the message bar of the window).

²⁴ Neither of these capabilities is not currently supported by Motif. However, implementation of a search capability is recommended, especially when users are presented with lists that contain a large number of items. A speed search capability is included in Windows.

6.7 SCROLL BARS

6.7.1 Appearance

Scroll bars are used to view textual or graphic information when it exceeds the space available to display it. Vertical scroll bars control backward and forward movement through the information; horizontal scroll bars control left and right movement.

Motif Only: Scroll bars have the default dimensions defined by Motif, except that vertical scroll bars have a width of 16 pixels and horizontal scroll bars a height of 16 pixels (per Kobara).

A scroll bar contains a trough region, a slider, and stepper arrows, as shown in Figure 6-9. The trough region is the background of the scroll bar and represents visually the length of the information that users can scroll. The slider represents the window through which users view the information. The relative position of the slider represents the relative position of the information currently displayed in the window. The stepper arrows scroll incrementally through the information and indicate the direction of the scrolling movement. Users can scroll to the top or the bottom of the information but not beyond.

Windows Only: When a window cannot be scrolled any further, the associated stepper arrow is dimmed.

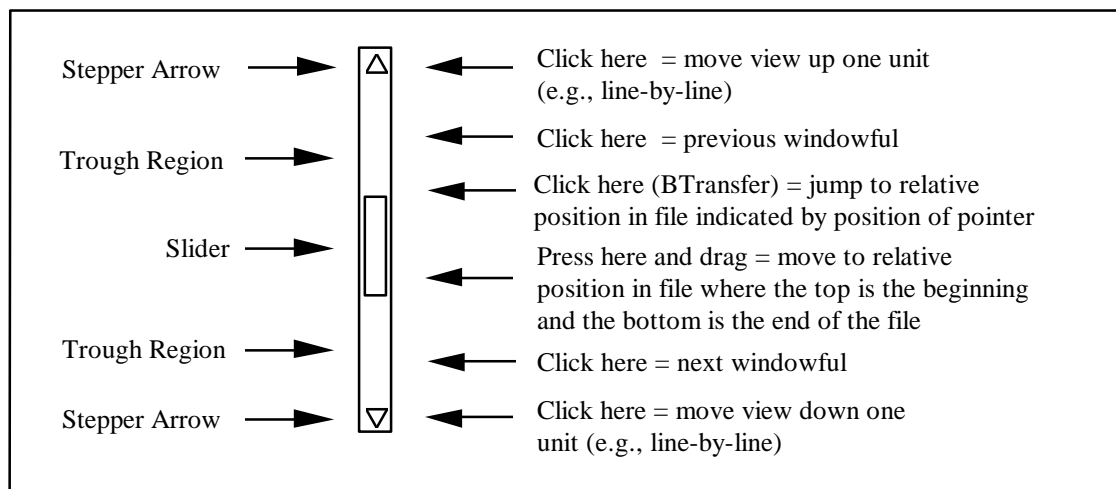


Figure 6-9. Components of a scroll bar.

6.7.2 Behavior

Pressing BSelect on a stepper arrow moves the slider in one-unit increments (e.g., one line or column) in the direction indicated by the arrow. Pressing BSelect on the trough region moves the slider one window length (or width) minus one unit (for overlap) at a time. Pressing BSelect and dragging the slider moves the slider in the pointer direction. Pressing BTransfer in the trough region moves the slider to the pointer position and then moves the slider in the pointer direction. <Esc> (or <Cancel> in Motif) returns the slider to its position before the sliding operation began.

When the scroll bar has focus, the arrow keys move the slider one increment in the arrow direction. <Ctrl> in combination with the arrow keys move the slider one large increment in the arrow direction. <PageUp>, <PageDown>, <Ctrl><PageUp>, and <Ctrl><PageDown> page in the specified direction. <Home> or <Ctrl><Home> and <End> or <Ctrl><End> scroll to the beginning and end of the scrollable region.

6.8 SCALES AND GAUGES

6.8.1 Appearance

A scale (as shown in figure 6-10) is used to select a value in a range. A scale consists of a trough region, a slider for selecting a scale value, and a label above or next to the slider showing the value currently set. A scale can also have arrow buttons which are used to select a scale value. The scale bar includes tick marks representing the range of available values and is labeled with the minimum and maximum values for the scale.

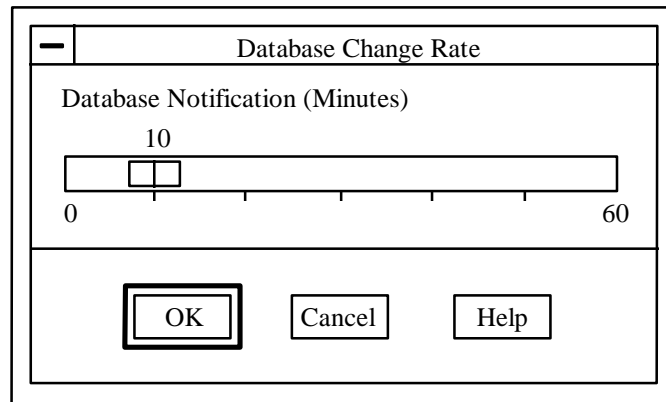


Figure 6-10. Example scale in Motif.

A gauge (shown in figure 6-11) is a display-only version of a scale for presenting values that users cannot change. For example, a working message window can include a gauge to provide dynamic feedback to users on the percent of a process that is complete. If a gauge is used to indicate processing, the trough region fills dynamically to indicate the relative amount of processing completed. If the exact percentage is important, the gauge includes a label indicating the current percentage value, and both the trough and label are updated dynamically. Because the values displayed by a gauge cannot be changed, a gauge does not include a slider or arrow buttons. However, a gauge is able to receive keyboard focus so that users can access Help for that control.

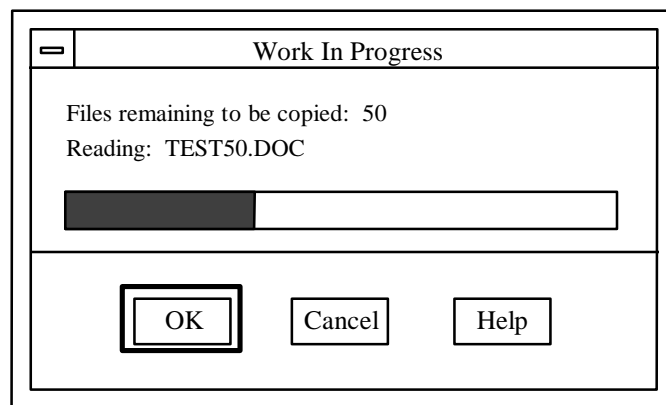


Figure 6-11. Example gauge in Motif.

6.8.2 Behavior

If the scale has stepper arrows, pressing BSelect on an arrow moves the slider one unit at a time in the arrow direction. Pressing BSelect on the trough region moves the slider one large increment (defined by the tick marks, if provided) at a time in the direction indicated. Pressing BSelect and dragging the slider moves the slider in the pointer direction. Pressing BTransfer in the trough region moves the slider to the pointer position and then moves the slider in the pointer direction. <Esc> (or <Cancel> in Motif) returns the slider to its position before the sliding operation began.

When the scale has focus, the arrow keys on the keyboard move the slider one increment in the specified direction. <Ctrl> in combination with the arrow keys move the slider one large increment, and <Home> or <Ctrl><Home> and <End> or <Ctrl><End> move the slider to the minimum and maximum scale values.

6.9 SEPARATORS

A separator is used to delimit the elements in a menu (e.g., groups of related options) or a window (e.g., the action and control areas). A separator does not support internal navigation.

Motif Only: A separator extends to the edges of the area it is delimiting (e.g., to the frame border in a window) and has an “etched in” appearance (per Kobara).

6.10 COMBINATION CONTROLS²⁵

6.10.1 Combo Boxes

A combo box consists of a text field and a list box displayed immediately below the text field, as shown in figure 6-12. A drop-down combo box (see figure 6-13) is a type of combo box consisting of a text field, a down-pointing arrow button, and a list box that is displayed when the arrow button is depressed. In both cases, users can either select one of the items from the list to display in the text field, or type directly in the field. When users select an item from the list, it replaces any text in the field. When users type in the text field, the text entered does not have to match one of the items in the list.

Windows Only: As users type in the text field, the list scrolls to the first item that begins with the characters being typed.

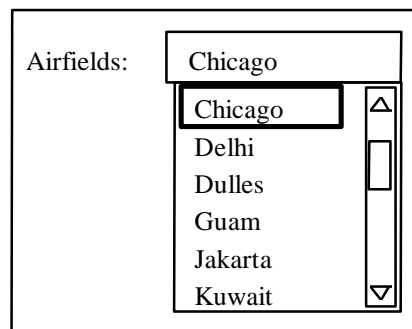


Figure 6-12. Example combo box in Motif.

²⁵ Combination controls are not available in previous versions of Motif. Windows distinguishes between combo boxes and drop-down combo boxes while CDE refers only to combo boxes. In addition, a combo box in CDE is the same as a drop-down combo box in Windows.

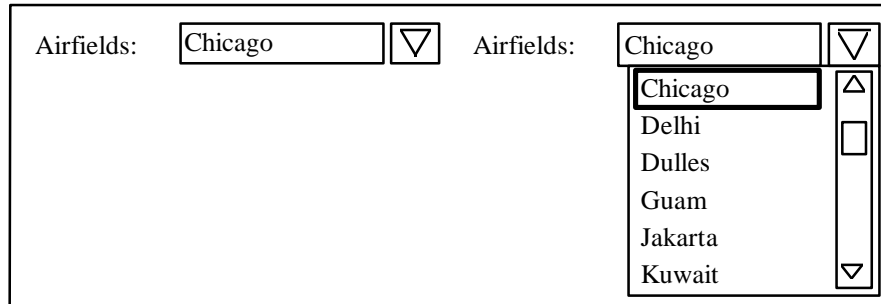


Figure 6-13. Example drop-down combo box in Motif.

The list in a combo or drop-down combo box is large enough to display six to eight items at a time, or all of the items if there are fewer than six. A vertical scroll bar is provided when the list is too long for all of the items to be visible. In a list that can be scrolled, the location cursor does not wrap. The combo box is wide enough that users can read all of the items in the list, with the text field the same width as the list. List items appear in sequential order based on nature of items and sequence expected. If it is appropriate for users to make no choice in a combo box, the list includes a “No Choice” item that, when selected, removes any text that is displayed in the text field. When a combo box is initially displayed, the text field can either be empty or pre-filled with a default item from the list. In the latter case, when the combo box receives keyboard focus, the default entry is highlighted so that text typed by users overwrites this text.

6.10.2 Spin Buttons

A spin button is used when users have to enter no more than 20 discrete, ordered values. The values can be numeric or a list of text strings (e.g., months of the year). A spin button consists of a single-line text field, with up- and down-pointing arrow buttons to the right of the field, as shown in figure 6-14. The text field can be editable (e.g., if the list of “spin” entries does not include all possible values) or noneditable (e.g., if the list is short and includes all possible values). When a spin button is displayed, the text field contains a default value. Users click on the up and down arrows to step through the values in the text field, and press on the arrows to step through the values continuously in the direction of the arrow. In both cases, when the largest or smallest value is reached, the entries wrap so that users can cycle continuously through the range of values. If the text field is editable, users can also type a value directly in the field.

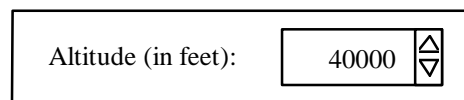


Figure 6-14. Example spin button in Motif.

If spin buttons are used to enter data such as date/time group or lat/long, separate spin buttons are used for each part of the entry. Spin buttons can be combined with standard text fields for data entry (e.g., separate spin buttons for month and day, with a text field for typing the year). If a value is typed in the text field, it is validated for correct syntax and format when users move focus out of the field.

6.11 STANDARD AND NONSTANDARD CONTROLS

6.11.1 Consistent Appearance and Behavior

If the application uses a nonstandard control, it has as much of the standard “look and feel” as possible. These features include a 3-D appearance for controls that can be manipulated by the user and

a similar color and shape coding scheme as other controls. If a control is a composite of existing controls, the composite has the appearance and behavior of the component controls.

All of the controls (both standard and nonstandard) in the application are identifiable solely on the basis of their appearance, and all controls with the same function have the same appearance. Users can distinguish between controls that are similar in shape (e.g., a push button and an option menu) on the basis of distinctive visual cues (e.g., an option menu should include a bar graphic). Users do not have to select a control in order to determine what it is and how it behaves.

6.11.2 Adapting Controls When Using Commercial Software

If the application uses commercial off-the-shelf (COTS) software, it configures these products to be compliant with the specifications presented here insofar as possible. In particular, existing controls are not modified in ways that conflict with the appearance and behavior defined in these specifications. For example, if a COTS product does not include a control such as a push button, the application does not include an object with the appearance of a text field and the behavior of a push button in order to provide this functionality. Likewise, the pull-down menus in a window are not altered so that the menu titles function as push buttons; this type of modification is inconsistent with both the normal behavior of menu titles (i.e., when selected, a pull-down menu is displayed) and with Motif and Windows guidelines on the arrangement of objects in windows (i.e., push buttons are not placed in a menu bar). When selecting a COTS product, consideration needs to be given to the extent to which the product can be adapted to fit the style defined by these specifications. If significant discrepancies (such as those described above) will result, the product may be inappropriate for use given the inconsistencies in “look and feel” that will result if it is integrated with other style-compliant applications.

7.0 SYSTEM AND APPLICATION DESIGN

7.1 CDE IMPLEMENTATION IN THE DII²⁶

7.1.1 The Desktop in DII Systems

DII systems installed on UNIX-based platforms use CDE to provide desktop management functions. The desktop configuration defined in this style guide is the default to be used by these systems unless operational requirements call for an alternative configuration (see section 1.5).

7.1.1.1 The CDE Front Panel

The functionality in the system is available from the Front Panel on the desktop. The default location of the Front Panel is near the bottom of the screen. The Front Panel can be moved by dragging the title bar or by using the move handles located along the sides of the panel; it can be minimized by clicking the Minimize button in the right corner of the panel.

The Front Panel provides access to various desktop managers (i.e., Application Manager, File Manager, Style Manager, Help Manager), generic utilities (e.g., Calendar, Printer, and Trash Can) and indicators (e.g., a clock), and workspace buttons for switching between available workspaces. The remainder of section 7.1 addresses the desktop management capabilities provided by CDE that are to be implemented in the system; Help Manager is addressed in section 11 on user support resources.

The system designates a default printer that is available to all applications. This printer is shown in the Front Panel, with information about other printers provided by the Print Manager application.

²⁶ A default desktop implementation for Windows-based systems will be addressed in a future version of this document.

Users can select any printer available to the system or use the default printer if they do not specify a printer name. They can submit a job for printing by using any of the Print features available on the desktop or by selecting a Print command from within the application.²⁷

7.1.1.2 DII Configuration

Modifying desktop functions. The system provides access to desktop and manager functions in a CDE-compliant manner unless the specifications presented here indicate otherwise. Individual applications determine the constraints for movement and deletion of objects created by the application, with users allowed to perform these actions if they are granted permission (i.e., access) to do so. If these actions are allowed, they are performed using the functionality provided by the desktop. If these actions are not allowed or do not apply, the corresponding features in the desktop are unavailable (i.e., are grayed out) when the application or file icons are selected; applications are not to alter how these functions are performed or remove them from the desktop.

Classification bar. The system provides a classification bar with markings indicating the current classification level.²⁸ The bar extends across the top of the screen and cannot be obscured by application windows when they are displayed or repositioned on the screen. The classification level appears in the middle of the bar; if desired, various status indicators (such as alerts) may be indicated at the left margin, and a digital date-time clock may be displayed at the right margin of the bar. Appendix D lists the colors to be used in the classification bar. All classification terms are presented in upper-case letters and spelled out, with caveats abbreviated in accordance with relevant security manuals and directives and with no embedded spaces within words in the label (e.g., SECRET, not S E C R E T).

7.1.2 Session Management

7.1.2.1 Starting and Ending a Session

All users are required to enter a valid identification and password before a desktop session is initiated. All users must successfully complete this login procedure before they can access any system functions.

Users can lock a system (e.g., to prevent unauthorized use when they are not present) by clicking on the Lock control in the Front Panel and to unlock the system by typing their password. In addition, users can configure the screen saver and lock features for the system using controls in Style Manager.

To end a session, users click on the Exit control in the Front Panel or choose the Log Out option in the Workspace menu. Based on their Logout preference setting in Style Manager, users may be prompted to confirm this action. When users log out of the desktop, the current session is saved so that when they log in again, the desktop appears as it did at the end of the previous session. When a session ends, all application windows are closed, and when logout is complete, a login window is displayed on the screen.

7.1.2.2 DII Configuration

Login. The system does not allow users to access a Terminal window unless this privilege has been granted to them. In addition, the Failsafe Session and Command Line Login features in CDE are

²⁷ Other desktop utilities will be addressed in a future version of this document.

²⁸ Compliance with the style guidelines for the Compartmented Mode Workstation, which is included as an appendix to the DoD style guide, will be addressed in a future version of this document.

removed. Finally, the system does not provide options to select a language other than English unless other languages are supported (see section 12 on internationalization).

Session. CDE provides two options with regard to session configuration. In the “current session” option, which is the default, the desktop saves session information (e.g., current appearance and behavior settings, applications that are running) when the user logs out and restores the system using this information when the user logs in again. In the “home session” option, users can explicitly save a particular session configuration and then choose to have that session restored whenever they log in. The system supports both of these options and allow users to select between them in Style Manager.

7.1.3 Application Management

7.1.3.1 Accessing Applications and Application Groups

The system uses Application Manager to provide access to the collection of applications available on the system. The applications can be represented as individual icons or arranged in groups which are represented by a container icon. Four application groups containing various tools and utilities are built into the desktop. Desktop_Apps includes desktop applications such as File Manager, Style Manager, and Calculator. Desktop_Tools contains desktop administration and operating system tools, System_Admin contains tools used by system administrators, and Information provides access to various help topics. Application Manager in the system includes these built-in application groups and the other applications and application groups to which the user has been granted access by the system.

Users can launch an application (and open an application window) either by double clicking on its application icon in Application Manager or pressing <Return> twice when focus is on the icon. In the case of applications that use or create data files, users can launch the application when they double click on one of its file icons in File Manager or when they drop the file icon on the application icon; in this case, the data file is loaded into the application window that is opened. Finally, the Personal Applications subpanel in the Front Panel provides access to applications that are used frequently by users. If users have dragged an application icon from Application Manager to this subpanel, they can launch the application by clicking on its control in the subpanel or by dragging one of its file icons from File Manager to the control.

Users cannot rename or delete an application icon unless they have been granted permission to do so. Users can launch only one instance of an application from the desktop; double clicking on the icon for an application that is running only raises the application window to the front of the screen.

7.1.3.2 DII Configuration

Built-in Applications. DII systems configure the set of utilities and tools available in the built-in application groups based on the privileges granted the user by the system. In particular, the Terminal Emulator application is not included in Application Manager for generic users, and the Terminal application is not available to them as a control in the Front Panel or from the File Menu in File Manager.

Installed Applications. Application Manager is the single repository for applications and application groups that the system adds to the desktop. DII applications are not installed in the Front Panel or its subpanels; only CDE components are installed there. However, if users choose to, they can drag frequently used applications to the Personal Applications subpanel and launch them from there. Only those applications which users have permission to execute are available in the Application Manager window; applications (as well as any files or folders used by the application) to which users do not have access are not visible anywhere on the desktop. Users can create new application groups within which to place the applications to which they are granted access.

Each application in the system is represented by a single application icon, with individual segments in the application available to users as defined in section 7.2. The label for the icon is the name of the application. Users can access all of the functionality in the application from this icon. A container defining an application group can be used instead of an application icon if other files (e.g., Read Me text, templates, sample data files) need to be bundled with the application. However, the application group includes an application icon that is used to launch the application. In addition, an application group may be used as the container for a suite of related applications (e.g., a Microsoft Office application group containing Word, Excel, and PowerPoint application icons).

7.1.4 File Management

7.1.4.1 Accessing Files and Folders

The system uses File Manager to display the files and folders available to users. Users can configure File Manager based on the way they want to see the content of folders (e.g., tree view, folder view); they can also change how objects are represented (e.g., by name, large icon, small icon, name, date, size) and the order in which icons are sorted (e.g., alphabetically, by file type, date, size). Opening a folder (e.g., by double clicking on it) changes the File Manager view to show the contents of that folder; opening a file starts the application that created the file and loads the contents of the file into the application window that is displayed.

Users can create new files and folders in the File Manager window. New folders use the standard icon provided by the desktop while new files use the icon defined by the application (see section 7.2.3.1 and appendix D). In both cases, user are required to name the object. Users can delete files and folders by dragging them to the Trash Can and then emptying the trash (i.e., choosing the Shred menu option and then confirming the action); in addition, the Trash Can is emptied when users log out of a session. Deleting a file or folder moves the icon from its location on the desktop (e.g., the File Manager window) to the Trash Can window; emptying the trash removes the icon from this window and permanently deletes the object from the system.

7.1.4.2 DII Configuration

Terminal access. File Manager allows users to open a Terminal Emulator window with the same current folder as the File Manager window. As indicated previously, DII systems are to limit Terminal access to those users with this privilege.

7.1.5 Workspace Management

7.1.5.1 Using Workspaces

The system uses Workspace Manager to control workspaces (e.g., the number available, which are open), workspace backdrops, and the Front Panel. Workspaces function as virtual desktops by increasing the amount of screen "real estate" available to users and creating customized areas where applications performing related functions can be placed. The system allows users to add, remove, and rename workspaces, to place applications in one, some, or all available workspaces, and to switch between workspaces.

A Workspace menu containing options for managing the workspace is available on all workspaces. Clicking BMenu on an empty part of the workspace displays the menu; access to this menu (i.e., to display the menu or choose a menu option) is not available from the keyboard. If any of the following workspace management functions are included in a Workspace menu, they are ordered: Shuffle Up,

Shuffle Down, Refresh, Minimize/Restore Front Panel, Restart Workspace Manager, and Log Out. These options execute the actions listed in appendix C.

7.1.5.2 DII Configuration

TED enhancements. The system implements workspace enhancements provided by the TED version of CDE. Graphical Workspace Manager provides a miniature view of the contents of each workspace in the user's session and allows users to navigate among these workspaces, move applications between workspaces, and change the stacking order of application windows in each workspace. Users are able to access this manager from either a control on the Front Panel or from the Workspace menu. In addition, the Application List option is available in the Workspace menu so that users can move easily between applications running on the current or a different workspace. Finally, a Workspaces control is added to the set of user preference features available for configuration in Style Manager.

Multiple monitors. The desktop implementation defined here assumes a default hardware suite that includes a single monitor. If the system uses multiple monitors, it configures the desktop to display a separate Front Panel on each screen, with each screen functioning as a unique desktop environment.

7.1.6 Style Management

7.1.6.1 Customizing System Style

Style Manager provides controls for desktop-wide customization of various features of system appearance and behavior. The system makes the following standard controls in Style Manager available to users:

Color - colors to be used in different areas of the screen (e.g., application windows, workspaces)

Font - font size to be used in window labels and text

Backdrop - pattern to be used to cover each workspace

Keyboard - volume of key clicks and rate at which characters repeat when users hold down their keys

Mouse - reversal of functions assigned to left and right buttons, assignment of the middle button to Adjust or Transfer functions, maximum time between clicks of a double click, and speed and threshold for accelerated pointer movement

Beep - volume, tone, and duration of auditory signals

Windows - focus policy, how active windows are displayed, and where window icons are displayed

Startup - session started at next login (e.g., current vs. home session), session to be the home session, and enable/disable logout confirmation

The system supports the Screen With Lock extension to the Screen control so that users can choose to enable or disable a screen saver and screen lock on timeout. In addition, the system implements a Workspaces control (see section 7.1.5.2) so that users can choose whether to synchronize workspaces on multiple screens (if they are using more than one monitor) and whether to hide or show workspace buttons.

7.1.6.2 DII Configuration

Color. Appendix D provides a group of color palettes that are appropriate for use in an office-like operational environment with normal ambient lighting; the Gray75 color palette is designated as the default. If the system is installed in this environment, it implements these color palettes (instead of the ones provided by CDE) in the Color control of Style Manager. If the system is installed in another

environment where these palettes may be inappropriate, it implements the DII default and defines other system-specific palettes that are tailored to this environment. Any alternative color palettes are available to users in the Color control in Style Manager.

The system allows the desktop to choose the number of color sets used in a palette, with users able to select from among those available. Palettes are configured to generate foreground color dynamically based on the background color selected. The system implements the centralized color management capability provided by the desktop so that applications change color dynamically when users select a different palette. Applications that cannot use dynamic colors use the DII default palette as their color set (see appendix D).

DII systems should be able to constrain the color selection options available in the Color control of Style Manager. In particular, users should be limited to the predefined color palettes defined by the system. Users should be able to choose from the Number of Colors setting for each of these palettes but not allowed to modify individual colors in a palette or add or delete palettes. This latter capability is not currently available in CDE but is an enhancement desired in a future release of the product.

Font. The Font control in Style Manager provides users with a choice of seven font sizes in which to display text. The system implements the centralized font management capability provided by the desktop, using the default system and user fonts defined by the platform on which the system is installed (see appendix D). Applications use the fonts for which font aliases are defined in CDE (see appendix E). DII systems should be able to provide users with a choice of fonts as well as font sizes; this capability is an enhancement desired in a future release of CDE.

Windows. The Windows control allows users to select between implicit and explicit focus. DII systems should be able to configure this control to restrict users to a single focus policy or to allow them to choose between them; the preferred implementation for DII systems is to restrict users to explicit focus only.

7.2 APPLICATION DESIGN IN THE DII

Two models are available for the design of applications in DII systems, one based on centralized task management (CTM) within the application and the other based on distributed task management (DTM). In these models, an application may consist of a single segment, a parent and one or more nested segments, or multiple independent segments, or it may access a segment as a shared resource.

7.2.1 Applications Providing Centralized Task Management

A CTM application is designed to perform a single primary task or multiple related primary tasks (see figure 7-1). One task in the application is designated as the default; the primary window for that task opens when the application is launched and provides application control (i.e., P(AC) in figure 7-1). Navigation among tasks (i.e., access to other primary windows) is available only from within the application.

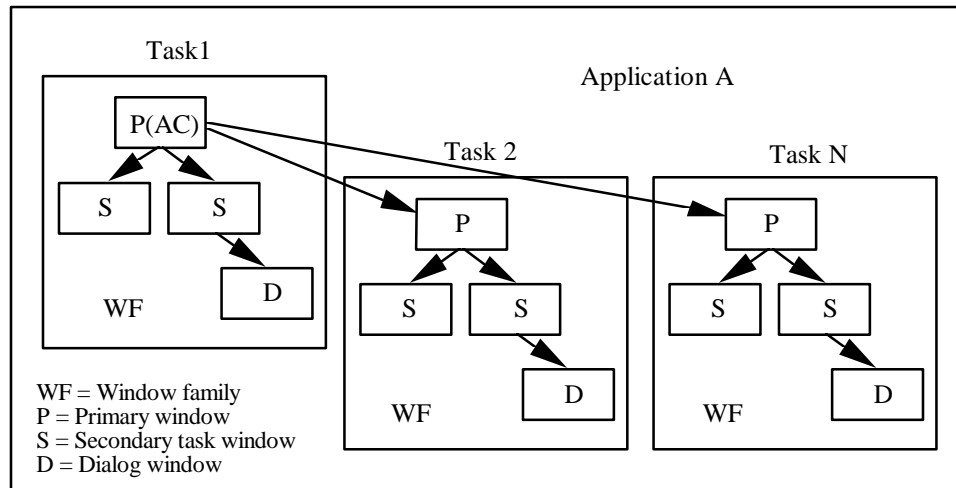


Figure 7-1. Design of a CTM application.

7.2.1.1 Basic Implementation

Launching. Launching the application (using methods listed in section 7.1.3.1) opens the window with application control, as shown in figure 7-2. Navigation among tasks in the application may be limited to only this window or also be provided in the other primary windows in the application.

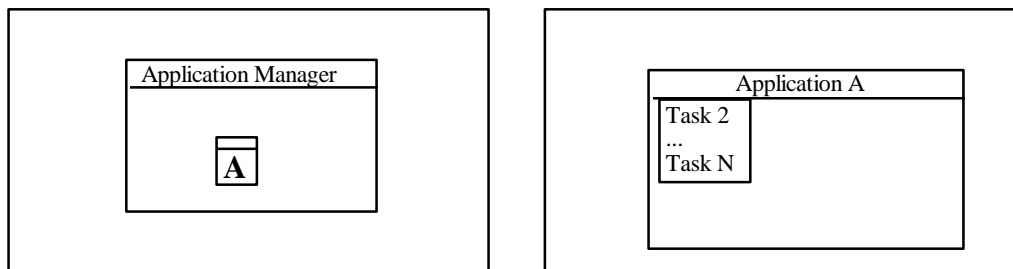


Figure 7-2. Launching a CTM application.

Closing and exiting. A Close option is available from the menu bar of each primary window in the application (including the one with application control); selecting this option closes all windows in the family parented by the window and quits processing in the window.²⁹ Primary windows without a menu bar include a Close push button that performs this action. An Exit option is available only from the menu bar of the window with application control; selecting this option closes all windows and quits processing in the application.

Window naming. The title of the window with application control is the name of the application; other primary windows are identified by application and task name, with no separator between the names, as shown in figure 7-3. Primary windows in which a file has been loaded are identified by application name, followed by a hyphen, and then the name of the file; the hyphen is preceded and followed by spaces. Secondary windows are identified by application name, followed by a colon, and

²⁹ Per version 1.2 of the *Motif Style Guide*: "When an application has multiple primary windows, the window manager should treat each primary window as if it were an independent application" (p 7-4). As a result, selecting the Close option in the application control window closes only the windows in the family parented by this window and not all of the windows in the application.

then the name of the secondary window; the colon is preceded and followed by spaces. The name of the application can be abbreviated if space is limited.

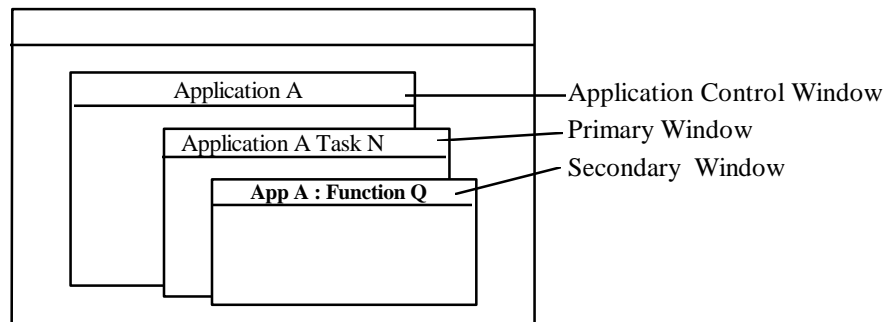


Figure 7-3. Window naming conventions in CTM applications.

7.2.1.2 Nested Implementation

The basic implementation of a CTM application assumes that the software for each task comes from a single source (i.e., one segment). Using different segments to perform the tasks in the application results in a nested implementation in which one segment is launched from within another (see figure 7-4). In this implementation, users would view the child segment as one of the tasks available within the parent segment.

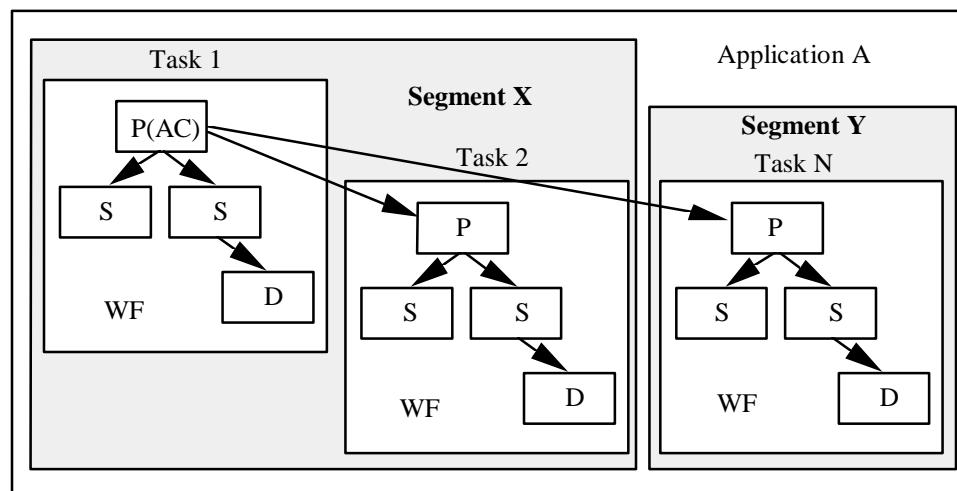


Figure 7-4. Nested segments within a CTM application.

In a nested implementation, the parent segment provides application control for the overall application. The application icon is named for the parent segment; double clicking on the icon launches this segment. Closing a primary window in the parent segment has no effect on windows in a child segment. Exiting the window with application control in the parent segment closes all of the windows and quits processing in both parent and child segments.

The child segment is not available on the desktop and can only be launched from within the parent segment. A Close option is available from the menu bar of the primary window(s) in the child segment; selecting this option closes all windows in the family parented by the window and quits processing in the window. An Exit option can be included in the primary window of the child segment, or it can rely

on the parent segment to provide this option. The name of the parent segment is used as the application name in the title of all primary windows in the child segment.

7.2.2 Applications Providing Distributed Task Management

A DTM application consists of multiple top-level tasks that can be opened and exited independently (see figure 7-5). Navigation among tasks is available from both the application icon on the desktop and within the application. One of the tasks is designated as the default for the application; the primary window for this task opens when the application is launched from the desktop. Application control is not assigned to the default window; instead, it is distributed among the primary windows in the application.³⁰

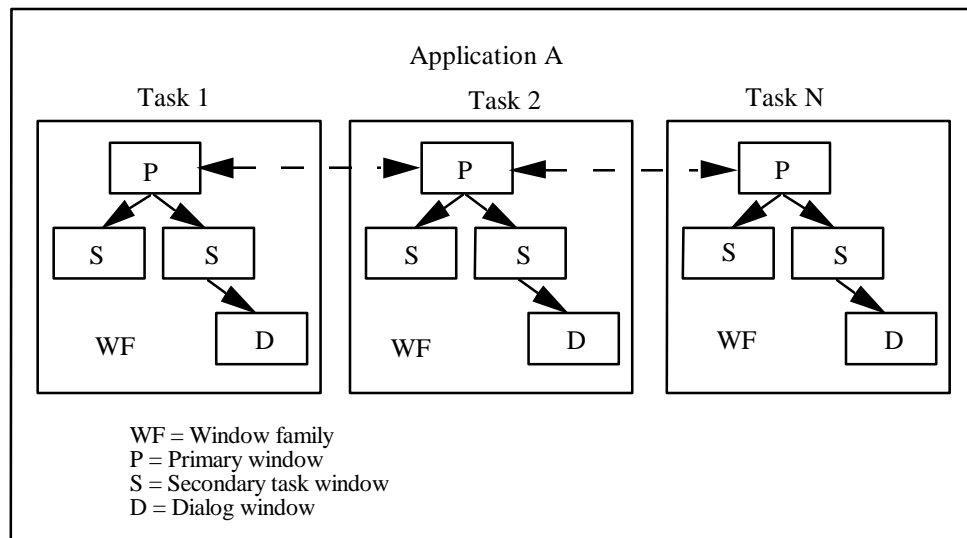


Figure 7-5. Design of a DTM application.

The tasks in a DTM application may be performed by software from a single source or from multiple sources. For example, a logistics application might support tasks related to data query, message generation, and data display, with the software for each task contributed by either the same or different segments. Because a DTM application consists of multiple independent tasks, the basic implementation defined here applies to both single-source and multi-source applications.

7.2.2.1 Basic Implementation

Launching. A pop-up menu is attached to the application icon listing all of the top-level tasks in the application (as shown in the upper part of figure 7-6). Double clicking on the icon launches the default task and open its primary window. Selecting one of the tasks from the pop-up menu launches the task and opens its primary window. Launching any of the tasks in the application changes the pop-up menu option for the task to unavailable. In addition, each primary window includes menu options from which all of the other top-level tasks in the application can be accessed. These menu options can be included in a single menu or distributed among several menus (as shown in the lower part of figure 7-6).

³⁰ This model differs from the approach defined in the Bellcore style guide for applications consisting of multiple primary tasks. In that document, each top-level task (i.e., window family) is represented by an icon on the desktop, with application control assigned to a single primary window in the application.

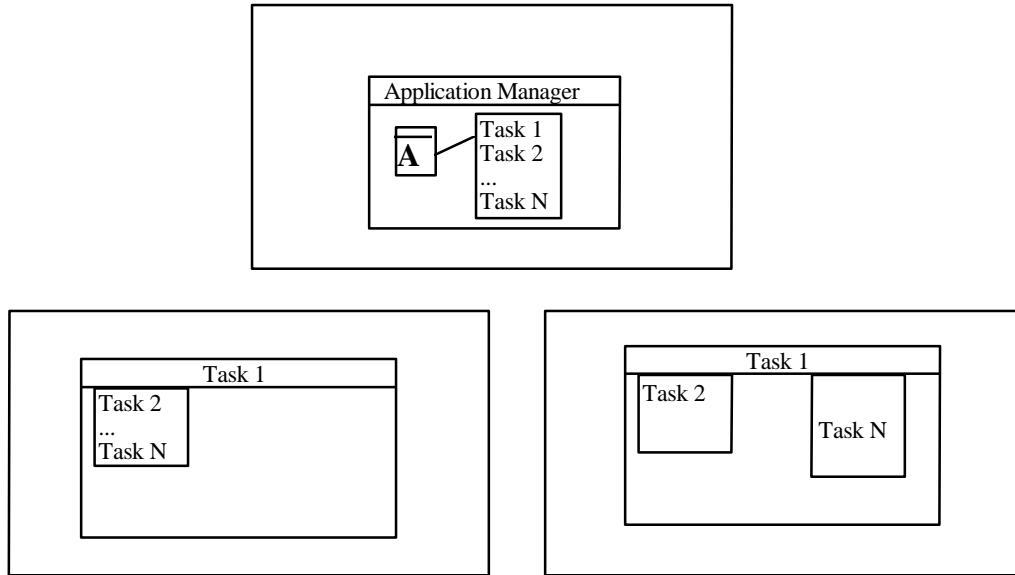


Figure 7-6. Launching a DTM application.

Closing and exiting. Close and Exit options are available from the menu bar of each primary window in the application. Selecting either of these options performs the same action, i.e., closes the windows in the family parented by the window and quits processing in the window (i.e., the task). Exiting the last primary window in the application also exits the application. An Exit All option can be included in each primary window so that users can exit each task separately or exit the entire application in a single action. When this option is selected, each primary window receives focus prior to being exited, with a prompt displayed if the window contains unsaved data. The last primary window exited also exits the application.

Window naming. Primary windows in the application are identified by task name, and secondary windows are identified by task name, followed by a colon, and then the name of the secondary window; the colon is preceded and followed by spaces (see figure 7-7).³¹

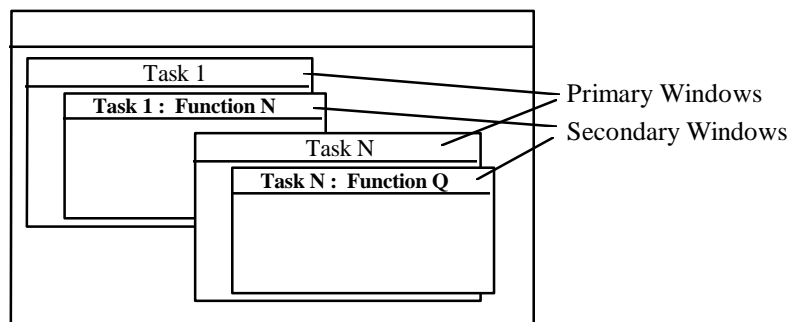


Figure 7-7. Window naming conventions in DTM applications.

³¹ Primary windows for each task are not linked to the application by name because the tasks in a given application can be performed by different segments, each of which may be available in more than one application. One disadvantage of not including application name is that users cannot easily identify the set of tasks that will be exited if they select an Exit All option in one of the windows.

7.2.2.2 Resource Sharing Among Applications

The DTM application model can be extended to support the implementation of resource sharing among applications. In this case, common support applications within a system provide generic services which users are able to access independently; these applications also make their services available to mission applications, either individually or as a shared resource. For example, Support Application A might provide basic chart functionality that can be accessed from the desktop. This functionality would also be available as a “task” within Mission Applications X and Y. Each application could open separate map windows (i.e., P(1) and P(2) in figure 7-8a) for plotting their data, or they could share a single map window in order to provide a common tactical picture (figure 7-8b).

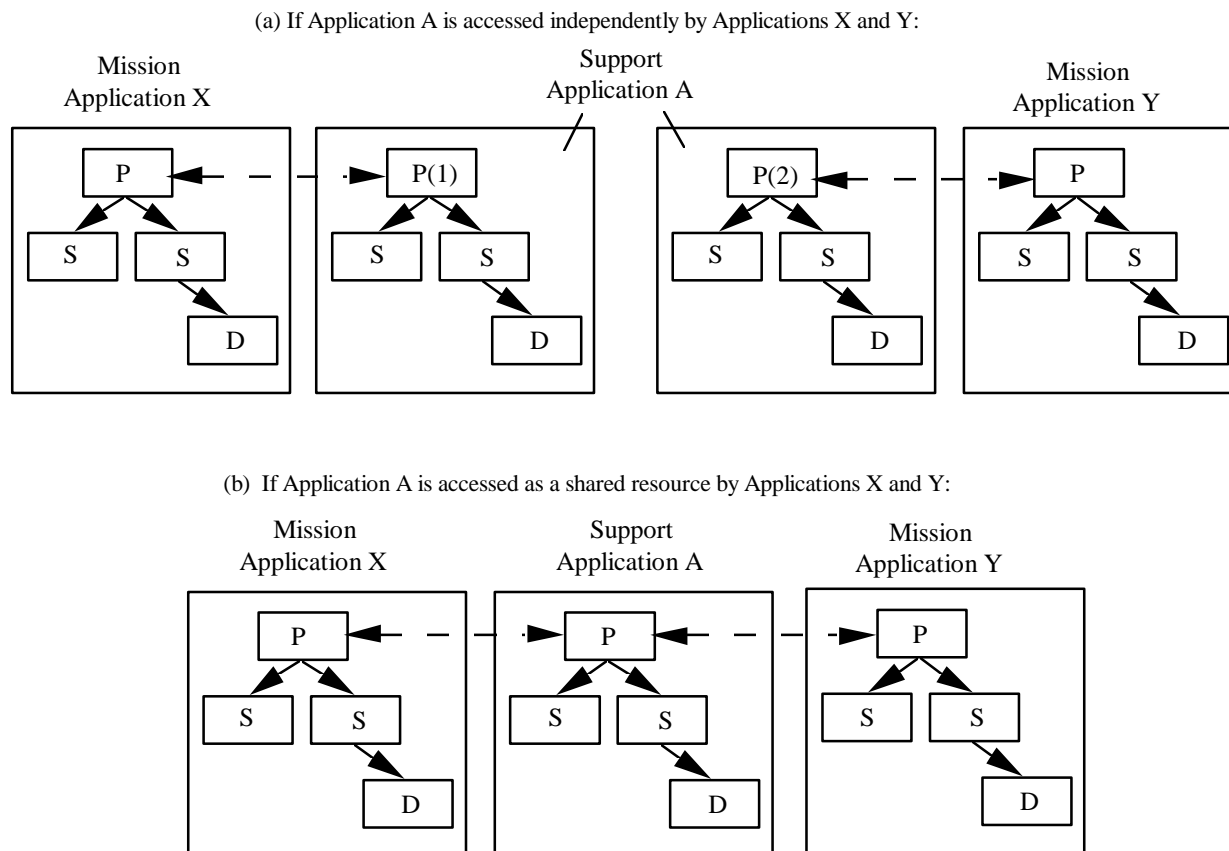
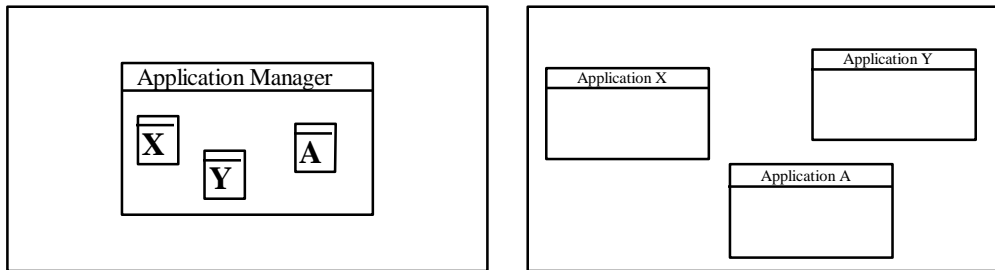


Figure 7-8. Providing independent or shared access to support services.

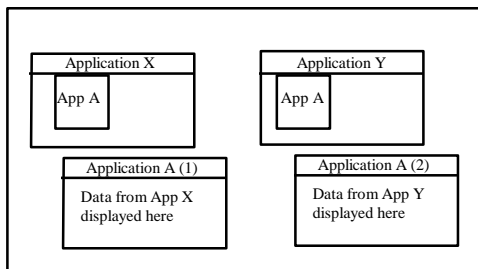
Launching Application A, X, or Y from the desktop opens the primary window for the application (see figure 7-9a). Launching Application A from a menu option within either Application X or Y changes the appearance of the option to unavailable; when separate windows from Application A are opened, each instance is identified by number (see figure 7-9b).³²

³² A weakness of this approach is that support application windows are not linked to the associated mission application by name (e.g., so that users can easily distinguish the map window displaying data from App X from the one displaying data from App Y).

(a) If Application A is launched as an independent application (similar to Applications X and Y):



(b) If Application A is launched individually from within Application X or Y:



(c) If Application A is launched as a shared resource by Application X or Y:

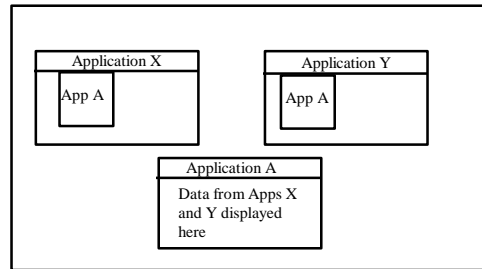


Figure 7-9. Options for launching a support application in a shared-resource implementation.

Each mission and support application is closed and exited independently; executing these actions in any of the applications affects only the windows in the application. When Application A is launched as a shared resource (see figure 7-9c) and then Application X or Y is exited, the shared window remains open but the data from the application being exited are removed from the window. When Application A is launched from within another application but is not shared, the latter application can provide an Exit All option to exit both it and Application A. When Application A is a shared resource, Exit All is not be available (since the shared window may contain data from an application other than the one being exited).

7.2.3 Other Design Considerations

7.2.3.1 Icon Design in CDE (Motif Only)

A key feature of CDE is the use of icons to represent both an application and the files it creates. Appendix D describes CDE requirements with respect to icon design. The applications provides the icon set in the three sizes and two formats indicated in this appendix.³³ All of the icons in the set are designed to a common theme (e.g., a similar graphic image), with individual icons tailored to reflect specific application functions. The application icon has a three-dimensional style and can fill the entire icon bounding box or be irregular in shape. The container icon (e.g., an application group) is designed so that its shape provides cues as to its function and the kind of objects it contains. Document and file icons indicate what application is associated with the document or file and the kind of data stored in the icon; if an application supports multiple file formats, document icons use the same basic file graphic but have a different “tag” to distinguish each output format. Finally, if the application may be used by non-English audiences, icons are designed to use international symbols whenever possible (see section 12 on internationalization).

³³ Additional direction with regard to icon design, including a standard set of icon graphics, will be provided in a future version of this document.

7.2.3.2 Accessing Segments Within an Application

In applications composed of multiple segments, the tasks in each segment can occupy all or part of an application menu, be available in a submenu, or be distributed across several menus. If menus become excessively long (and extend beyond the bottom of the screen), the application uses submenus, following guidelines in section 5.5, to reduce menu length. Menus that cannot be shortened include controls (e.g., arrow buttons for scrolling) for users to view and select the options that extend beyond the bottom of the screen. The application determines when users can and cannot access each of the tasks performed by a segment and dims the associated menu option to indicate its unavailability.

8.0 APPLICATION WINDOW DESIGN

8.1 PRIMARY AND SECONDARY TASK WINDOWS

8.1.1 Window Components

A primary task window contains a window frame with resize borders, Window menu, Minimize, and Maximize buttons, and a title bar.

Motif Only: The Window menu includes (in this order) Restore, Move, Size, Minimize, Maximize, Lower, Occupy Workspace, Occupy All Workspaces, Unoccupy Workspace, and Close options.

Windows Only: The Window menu includes (in this order) Restore, Move, Size, Minimize, Maximize, and Close options and may include Switch To and Next options.

The area inside the window frame of a primary task window contains a title and a main area. The window also includes a menu bar and may have a message bar, as shown in figure 8-1.

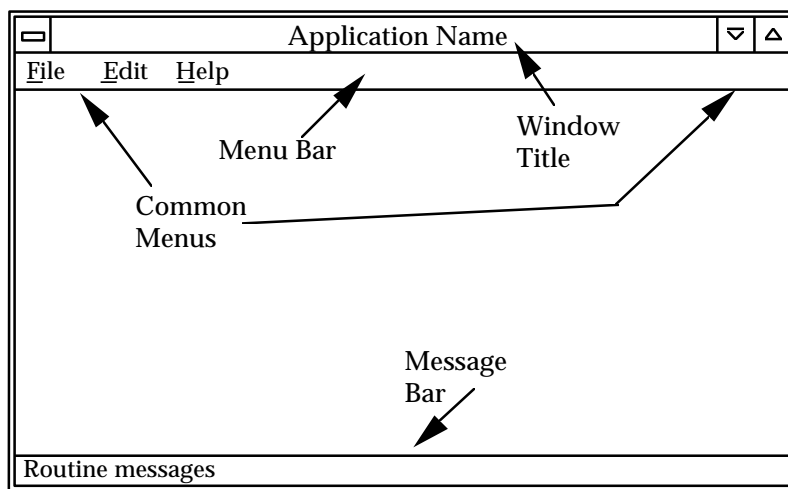


Figure 8-1. Example primary task window in Windows.

A secondary task window contains a window frame with resize borders, a Window menu button, and a title bar; the window may also contain Minimize and Maximize buttons.³⁴

³⁴ Although Motif and Windows indicate that secondary windows do not have a Minimize button, the specifications presented here allow this component to be present in order to provide greater flexibility in window management (e.g., so that users have easy access to window functionality without obscuring critical information in a primary window).

Motif: The Window menu in a secondary task window includes (in this order) Restore, Move, Size, Minimize (if available), Maximize (if available), Lower, and Close options.

Windows: The Window menu in a secondary task window includes (in this order) Restore, Move, Size, Minimize (if available), Maximize (if available), and Close options.

The area inside the window frame of a secondary task window contains a title and a main area. The window includes either a menu bar at the top or an action area at the bottom, as shown in figure 8-2; the window may also have a message bar. The window includes a menu bar if the number of user operations is more than five and/or if access to File and Edit operations is needed. The window includes an action area if the number of user operations is five or less and/or if access is needed to application data but not to File or Edit operations.

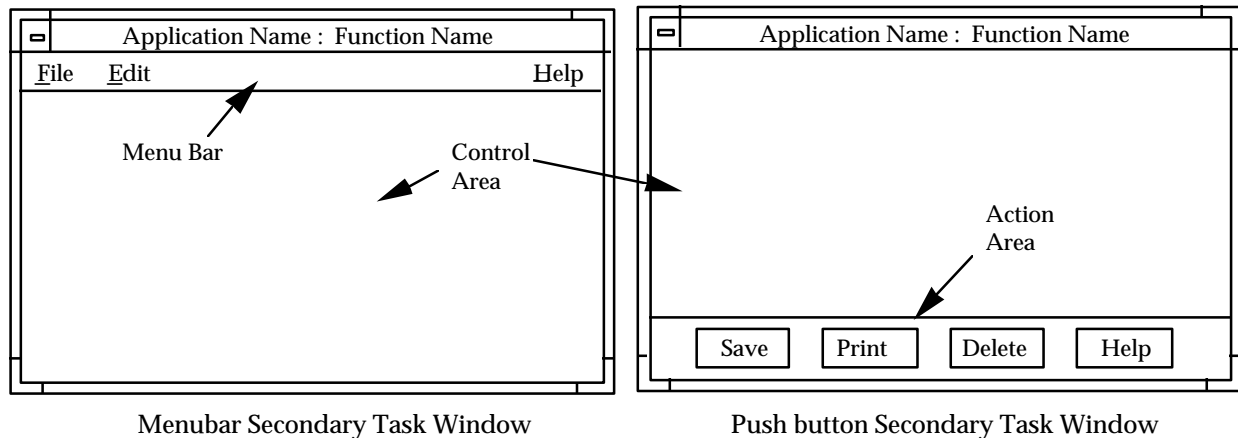


Figure 8-2. Example secondary task windows in Motif.

8.1.2 Window Design Guidelines

8.1.2.1 Window Title

The window title is centered in the title bar and presented in mixed case, with the first letter of each word capitalized. If a file name is included in the title, it is in mixed case in Motif and in upper case letters (including any extensions) in Windows. In both cases, a placeholder name (e.g., Untitled Document 1) is used if the file has not yet been named. Specifications concerning window naming are presented in section 7.2 on application design.

Each window title in the application is unique. The window title does not contain information such as the version of an application or the full path name for a file. In addition, the window title is not used to present dynamic information such as messages to the user. If selecting a menu option causes a secondary window to be displayed, the title of the window matches or refers to the wording of the option that displayed it.

8.1.2.2 Menu Bar

If a window includes a menu bar, it appears below the title bar and contains no more than ten menu titles plus Help.

Motif Only: The menu titles begin at the left margin of the menu bar and extend rightward, with Help at the right margin of the menu bar. The titles are displayed in buttons of equal size, using the default dimensions defined by Motif, except for margin width which is 8 pixels (per Kobara).

Windows Only: The menu titles begin at the left margin of the menu bar and extend rightward, with Help as the last menu. Help is placed next to the menu that precedes it.

The space between menu titles is sufficient (at least three character widths) so that multi-word titles can be distinguished from single-word titles. Commands (e.g., push buttons) are not included in the menu bar.

8.1.2.3 Common Menus

Motif and Windows conventions concerning menu design and content are followed except as needed to provide access to application-specific functions. If any of the following common menus is used, they are ordered: File, Edit, View, Options, Window (Windows only), Help. Application-specific menus can be inserted between these common ones, except as indicated below.

File menu. The first (i.e., leftmost) menu in the menu bar contains options for users to work with the data in the window as a whole. The title of this menu is File or an application-specific term with comparable meaning.

Motif Only: If the File menu includes any of the following options, they are ordered: New, Open, Save, Save As, Print, Close, and Exit. Separators follow the Open, Save As, and Print options.

Windows Only: If the File menu includes any of the following options, they are ordered: New, Open, Close, Save, Save As, Print, Print Setup, and Exit. Separators follow the Save As and Print Setup options.

Windows Only: If the File menu includes a list of most recently used files, the list precedes the Exit option. Selecting an option with a file name opens a window containing the file. If the file is already open, selecting the option raises that window to the front. The number of files in the list can range from three to eight but remains constant within the application. When a file is opened, the file name is placed at the top of the list in the menu and given the number 1 (e.g., 1 TEST.DOC) which serves as its mnemonic. When another file is opened, it is added to the top of the list (and given the number 1) and the previously opened files are renumbered and move down in the list.

Edit menu. If an Edit menu is present, it contains options that enable users to modify the data in the window. If File and Edit are both included, they are next to each other in the menu bar.

Motif Only: If the Edit menu includes any of the following options, they are ordered: Undo, Cut, Copy, Copy Link, Paste, Paste Link, Clear, Delete, Select All, Deselect All, Select Pasted, Reselect, Promote. Separators follow the Undo, Paste Link, and Delete options.

Windows Only: If the Edit menu includes any of the following options, they are ordered: Undo, Repeat, Cut, Copy, Paste, Paste Special, Clear, Delete, Select All, Find, Replace, and Links. Separators follow the Repeat, Select All, and Replace options.

View menu. If a View menu is present, it contains options for changing the user's view of the data but does not actually change the data. This menu can also contain options for controlling the display of interface elements such as toolbars.

Options menu. If an Options menu is present, it contains options for customizing the application.

Window menu (Windows Only). If a Window menu is present, it contains options for manipulating document windows. The menu includes New Window and window arrangement options (Tile, Cascade, or Arrange commands) and a list of open windows (similar to the list of most recently used files in the File menu). Selecting a window name from the menu raises that document window to the front and gives it focus. The active window is indicated by a check mark preceding the window name in the menu. The list can contain up to nine window names; if more than nine windows are open, the list includes a More option that, when selected, displays a dialog window with the names of all open document windows from which users can select.

Help menu. The Help menu provides access to additional information about the window or the application. See section 11.4 for additional information on application-level help.

Motif Only: If the Help menu includes any of the following options, they are ordered: Overview, Index, Table of Contents, Tasks, Reference, Tutorial, Keyboard, Mouse, Mouse and Keyboard, On Item, Using Help, and About.³⁵ The minimum set of options included in the menu is Overview, Tasks, Reference, On Item, Using Help, and About <application>. In addition, if the application provides unique function keys or accelerators, a Keyboard option is also included in the menu.

Windows Only: If the Help menu includes any of the following options, they are ordered: Contents, Search for Help On, Tutorial, How to Use Help, and About.

8.1.2.4 Arrangement of Controls

The main area of a window is organized into subareas based on the nature of information being presented or the type of action required by users.³⁶ Controls that perform similar or related functions are grouped together and surrounded by a frame, as shown in figure 8-3.

Motif Only: The frame has an “etched in” effect in order to be clearly different in appearance from a window control; a frame does not have a “shadow in” or “shadow out” effect since these effects are used for window controls (e.g., a text field or a push button). Frames include a margin (i.e., have a non-zero value for margin height and width) in order to provide sufficient space between the frame and the objects within it (per Kobara).

³⁵ These Help menu options supercede the two help models available in previous versions of Motif.

³⁶ The design of expandable windows will be addressed in a future version of this document.

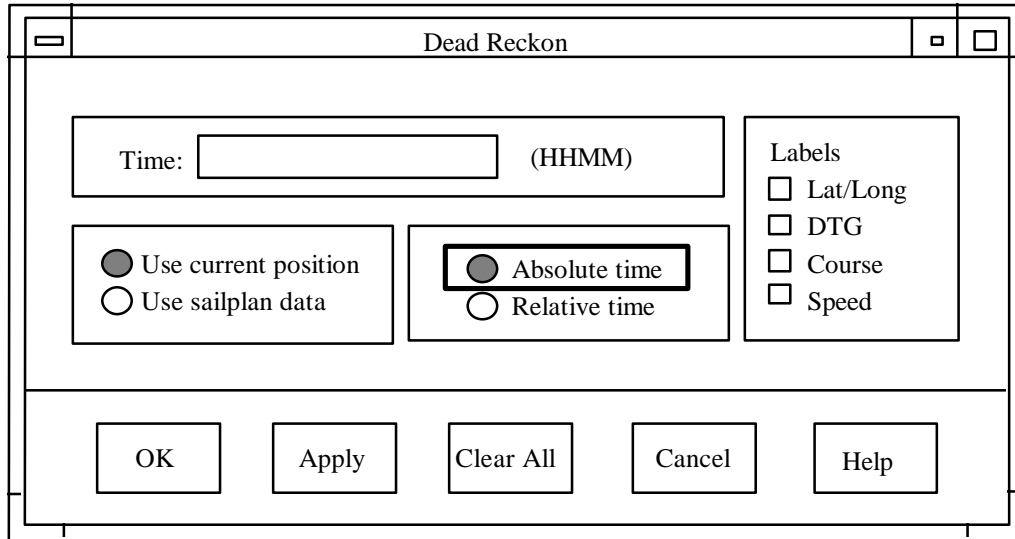


Figure 8-3. Example arrangement of control groups in a Motif window.

If a group of controls includes a heading, it is a label describing the function performed by the controls. The heading is placed either inside the frame or in the frame. If the heading is longer than the text in the controls, the size of the frame is extended so that it is wider than the heading. The heading is either left justified or centered within the frame; it is presented in mixed case, following normal capitalization rules, and is not followed by a colon.

The controls in a group are arranged in one or more rows or columns within the frame. The preferred orientation for a group of radio or check buttons is vertical and left-aligned; if placed horizontally, space is sufficient (at least twice the distance between the button and its label) so the button is paired with the label on the right, not left.

Motif Only: Groups of radio and check buttons use the default dimensions defined by Motif, including the value for spacing between the indicator and label.

When a window is initially displayed, all of the controls in the window reflect the current state of the application. For example, a window that allows users to change the font size of text in the application is displayed with the current font selected. If there is a preferred or expected choice within a window, a default option is defined within a control (e.g., a list box) or group of related controls (e.g., a set of check buttons), and this choice is selected (i.e., highlighted) when the window is displayed. If the expected choice cannot be anticipated, then a default is not designated. Controls that become unavailable are dimmed and not available for selection. Controls that are never available to users (e.g., if access to them is password controlled) do not appear in the window. When one of the controls in a window is selected but not executed (e.g., users choose the Cancel push button in the window), the selection is not saved and the control reverts to its original state when the window was first displayed.

8.1.2.5 Availability of Scroll Bars and Window Panes

If a scrollable area in a window is too small to view all of the contents, scroll bars are provided to allow scrolling of the area. The scroll bars are located to the right or at the bottom of the area. When scroll bars are provided, they scroll the main part of window only and not the menu bar or message bar in the window.

Motif Only: A scrollable window always displays its scroll bars regardless of whether the viewable window is the same size or smaller than the underlying data. The scroll bars are positioned 4 pixels from the viewable area in a scrolled window (per Kobara).

Windows Only: Scroll bar(s) remain displayed even if they become inactive.

Window panes are used to separate control areas when space is limited or to present two simultaneous views of the same data in a single window. Panes can be arranged either horizontally or vertically in a window.

Motif Only: Separators and sashes serve as the boundary between panes. Users resize panes by dragging the boundary between the panes using BSelect or BTransfer or by moving the sash using the arrow keys; making one pane larger makes the other pane smaller but does not affect the overall size of the window. Per Kobara, spacing in a paned window is set to the default of 8 pixels so that the sash and separator between panes remain displayed. In addition, sash height is set to 6 pixels, and sash width is either the same as scroll bar width or set to 12 pixels if no scroll bars are used. Finally, the distance the sash is indented is the same as the dimension by which the vertical scroll bars are offset from the right edge of the window; if there are no scroll bars, sash indent is set to -4, which places the sash close to the right edge of the pane.

Windows Only: Split boxes are used to divide the window into panes and then adjust the size of the viewing areas. A split box is a solid box located at the top of a vertical scroll bar or at the left end of a horizontal scroll bar. Users can drag the split box to divide the window into separate panes; double clicking on the split box divides the window in the middle. A split bar serves as the boundary between the panes. Dragging the split box or split bar to either end of the window closes the pane in the direction of the drag. When a window is split, scroll bars are displayed (along with the split bar) so that users can scroll each pane (i.e., perpendicular to the direction of the split) independently within the window.

8.1.2.6 Arrangement of Push Buttons

Motif Only: Push buttons are displayed horizontally, centered at bottom of the window, and separated from other controls with a separator, as shown in figure 8-3. Push buttons are ordered from left to right based on the sequence in which they will be used, with the most frequently used button on the left. Buttons indicating positive actions are at the left, followed by buttons indicating negative actions and canceling actions. A Help push button is included in every window and is the rightmost button.

Windows Only: Push buttons can be placed along the right margin or across the bottom of a window; the preferred implementation in the DII is the latter. Buttons that initiate actions (e.g., Find, Save) are placed at the left, followed by other push buttons (e.g., "GoTo," "GoSub"), and then Help. If there is an OK button, it is placed first (even if it is not the default) and followed by Cancel, with both separated from the other action buttons. If there is no OK button, Cancel follows the other action buttons but precedes the "GoTo," "GoSub," and Help buttons.

Push buttons appear in the same order throughout the application. A window contains no more than seven push buttons, including Help. Close and Cancel are not included as push buttons in the same window.

Windows Only: Windows that support multiple actions contain a Cancel push button when initially displayed; if the actions performed in the window make irreversible changes to data, the label of the Cancel button changes to Close as soon as the first such action is executed. When the window is closed and then reopened, the button label reverts to Cancel.

If users can perform multiple actions within a window and these actions can affect different objects or data elements in the window, push buttons are labeled to reflect the object(s) that each button affects and are located near the object(s) to which they relate. Push buttons related to overall window functionality (e.g., OK, Cancel) are placed along the bottom of the window.

Motif Only: If users can perform mutually exclusive actions (e.g., Pause and Resume) in a window, the window contains a push button for each action, and the label of the button that is not available at the time is grayed out. The window does not contain a single push button whose label changes to indicate the action available.

Windows Only: If the function performed by a push button changes depending on the state of the application, the label changes accordingly. A single push button, rather than separate buttons, is used in the window.

Windows Only: When a push button with an ellipsis in its label is selected, a dialog window requesting additional information is displayed. The parent window containing the button can close when the child window opens (i.e., a “GoTo” window) or the parent window can remain open (i.e., a “GoSub” window). In both cases, if the child window can also be opened as a result of a menu selection, the window title is the same as the menu option. If users open a “GoSub” window, make changes in the window that cannot be undone, and then return to the parent window, selecting Cancel in the parent cancels all changes made in both the parent and child windows.

8.1.2.7 Default Push Buttons

If a default action is available, it is the action that users are most likely to execute in the window. If an expected choice cannot be anticipated, there is no default defined for the window; <Enter> (or <Return> in Motif) has no effect, and users must select one of the available push buttons to execute an action.

Motif Only: When a window is initially displayed, the default push button is the leftmost button in a group of push buttons.

The same button is the default whenever the window is displayed. The default designation may be assigned to a different push button depending on the control that has keyboard focus in the window. When keyboard focus is on a push button, its action is the default, and it is shown with default highlighting. This highlight moves with the location cursor during keyboard navigation in a group of push buttons and returns to the original button when focus leaves the push button group. If the default action in a window varies, one push button always shows the default highlighting except when there is no default action currently available. If focus is outside a window, the highlight is displayed on the push button whose action will be the default when focus returns to the window.

When more than one action is available in a window, the default push button is the nondestructive one. If the default highlight moves to a push button other than the one originally identified as the default, the highlight reverts to the original button when the window is closed and then opened again. The action performed by the default push button is reversible (e.g., by selecting an Undo menu option or a Cancel push button).

8.1.2.8 Tool Bars

If a tool bar is included in a window, it provides redundant access to functionality available elsewhere in the window (e.g., in a pull-down menu). For example, the buttons in a tool bar can invoke modes (e.g., different drawing tools), apply settings (e.g., select font style or size), or execute actions (e.g., invoke a print command).

Motif Only: A tool bar is used only in windows with a menu bar; if present, the tool bar is located at the top of the window, below the menu bar, as shown in figure 8-4. The tool bar is the same width as the window and the same height as the menu bar.

Windows Only: A tool bar can occupy a fixed position within a window (e.g., below the menu bar, along the left margin or bottom of the window), or it can be placed in a separate dialog window and be movable. In the latter case, the tool bar is always displayed in front of the window to which it applies. The dialog window with the tool bar includes a title bar (so the window can be dragged to a new location) and a Window menu with Move and Close options.

A window with a tool bar also includes a message bar at the bottom of the window so that information about the purpose of a button in the tool bar can be provided when the pointer is on the button or the button has focus. Users have the option to hide or show the tool bar (e.g., from a toggle-type option in the View menu).

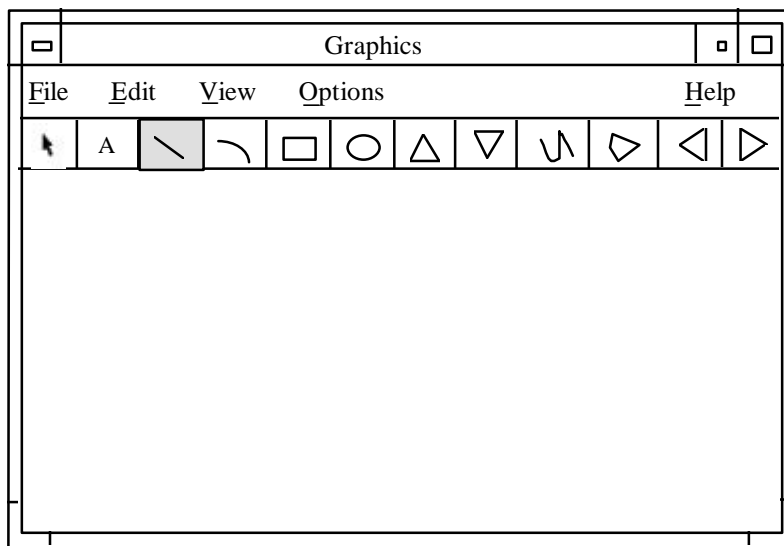


Figure 8-4. Example tool bar in Motif.

Motif Only: A tool bar contains no more than 20 action buttons. The buttons are of equal size and evenly spaced across the tool bar. They are arranged in an order expected by users; if one does not exist, they are arranged by frequency of occurrence, sequence of use, or importance.

Windows Only: A toolbar can contain drop-down lists as well as groups of action buttons; spacing is 12 pixels between the lists and buttons and 6 pixels between button groups. A toolbox (i.e., a toolbar placed along the left margin of a window) contains groups of action buttons arranged vertically, with no space between the buttons.

The normal appearance of a button is raised; when selected, the button is recessed and changes appearance (e.g., highlights) to indicate its selected state. A button that becomes unavailable changes appearance (e.g., is grayed out) to show that it cannot be selected. Whenever a menu option becomes unavailable, the corresponding button in the tool bar is also shown as unavailable.

The labels for buttons in a control bar should be presented as icons, with graphics that are the same size in each button.

Motif Only: Button size is 24 x 24 pixels.

Windows Only: Normal button size is 24 x 22 pixels, with a graphic image 16 x 15 pixels. If space is limited, button size is 22 x 18 pixels, with a graphic image 16 x 12 pixels. The graphic image is a black outline, with white fill within the outline where needed.

The icon represents a verb (rather than a noun) and can depict a before and after representation of the action (e.g., a small and large version of an object, connected by an arrow, to indicate Magnify), the tool that would accomplish the action (e.g., a pair of scissors to indicate Cut), or the action itself (e.g., a paintbrush filling an object with color to indicate Paint). One of these schemes is selected, and all of the icons in a tool bar designed to fit this scheme. Text labels describing the action executed by the button may also be included with the icon graphics.

Windows Only: If text is included with the graphic, the size of the button is larger and dependent on text length. The graphic is placed either above the text or to the left of the text; in the latter case, the button has the same height as a normal push button (i.e., 22 pixels), with 4 pixels between the graphic and text.

When a button that invokes a mode is selected, the button remains selected (i.e., highlighted) as long as the mode is in effect. The pointer shape changes to indicate type of operation users can perform while in the mode. The pointer has this shape whenever it is in the window where the mode is in effect. If users move the pointer outside this window, the pointer changes to the appropriate shape. A tool bar either provides a button for returning to an unselected state (i.e., exiting the mode) or automatically returns to an unselected state after an action is executed.

8.1.2.9 Message Bar

If a window includes a message bar, it is used to provide noncritical application messages to users, present simple help, or indicate status pertaining to the window. When the message bar is used to indicate status, a progress message is displayed when the action is initiated (e.g., "Drawing map...") and updated when the action is completed (e.g., "Drawing map...Done"). The text is removed from the message bar within 5 sec of action completion.

A message bar displays read-only text; users cannot type or modify any information presented in this area.

Motif Only: The message bar can have the same appearance as a noneditable text area (as was recommended in previous versions of Motif); the preferred implementation indicated in CDE is to widen the margin area at the bottom of the window so that messages can be presented there.

Windows Only: A status bar is a more elaborate version of a message bar in which messages as well as information about the current state of the application (e.g., cursor location, modes in effect) are displayed. If a status bar is used in a window, the application provides users with a way to hide or show the bar (e.g., as an option in the View menu) as desired.

8.1.2.10 Draggable Objects in Windows (Motif Only)

As indicated in section 3.5.1.1, the application is to provide a drag and drop capability for all objects represented as icons and for all window elements that users can directly manipulate. This capability provides redundant access to functionality available elsewhere (e.g., buttons, menus, dialog windows) in the application.

An icon graphic is included in a window to indicate that it contains a draggable object. The graphic is the same as the one used to represent the object in the File Manager on the desktop. The icon is placed next to any display of the contents of the object if one is present in the window. If there is no

such display, the icon is placed in the upper right corner of the window. The icon is 32 x 32 pixels and includes a label describing the kind of object the icon graphic represents. The graphic is also used as the source indicator in the drag icon.³⁷

8.1.2.11 Pop-up Menus and Text Fields in Windows

Pop-up menus are provided for those functions or objects in a window for which redundant access would improve task performance. The availability of these menus is limited to frequently executed functions or frequently selected objects included in a window. For example, pop-up menus with edit commands can be provided for the text fields in a window so that users do not have to move the pointer to and from an Edit pull-down menu. Similarly, a pop-up menu containing frequently executed actions can be available in a window so that users do not have to move the pointer to and from the push button area of the window.

Windows Only: Read-only pop-up text fields are used to display additional information about the text in a field when space within a window is limited. The presence of a pop-up field is indicated by underlining the word(s) in the text to which it relates. Clicking BSelect on the underlined text displays the pop-up field; clicking anywhere outside the field dismisses it. When the field is displayed, it is placed so that its top left corner is at the same position as the top left corner of the original text.

8.1.2.12 Mnemonics and Accelerators in Windows

Windows Only: Mnemonics are available as an additional method for keyboard navigation among controls in a window. When mnemonics are implemented, they behave as indicated in section 3.4.2 and 5.5 and use the characters listed in appendix C. The OK and Cancel push buttons do not have mnemonics, given that <Enter> activates the OK button if it is the default and <Esc> activates the Cancel button.

The use of accelerators in windows is limited to the set of function keys listed in appendix A. Accelerators are not assigned to individual controls since this functionality would not be visible in the window (i.e., part of the label for the control).

8.1.2.13 Document Windows (Windows Only)

A document window has the same window components as a primary window, and its Window menu contains the same options as the Window menu in the application window. The title of a document window is the name of the document and presented in mixed case.

Document windows appear within the borders of the application window. When a document window is maximized, the window is closed, and the data from the window are displayed in the application window. The title of the application window changes to include the document name (i.e., application name, followed by a hyphen, and then the document name), the Window menu button from the document window is displayed at the left end of the menu bar in the application window, and a Restore button for the document window is added to the right end of the menu bar. In addition, the application window has a scroll bar if the document window requires scrolling.

If the application has a menu bar, it appears in the application window, along with any controls (e.g., tool bars) that apply to all document windows. In applications that support different document

³⁷ The implementation of drag and drop for attachments will be addressed in a future version of this document.

types, the content of the menu bar can change depending on the document that currently has focus. Menu titles can be temporarily removed from the menu bar if no documents are open; the only menu titles included in the menu bar are those that can be executed when no documents are open.

8.2 DIALOG WINDOWS

8.2.1 Window Components

A dialog window contains a frame, a Window menu button, and a title bar; the window does not have resize borders or Minimize or Maximize buttons. The Window menu in a dialog window includes (in this order) Move and Close options.

Windows Only: If a dialog window is not movable, it does not have a title bar, and Move is not an option in the Window menu.

The area inside the window frame of the window contains a title and a main area. The main area includes a control area for presenting messages or controls and a push button action area at the bottom of the window for executing actions.³⁸

Motif Only: Message windows are modeless whenever possible.³⁹ An error message window is modal only if it is critical that users acknowledge having read the message prior to continuing to interact with the application.

Windows Only: Message windows are modal. Critical message windows are system modal while information and warning message windows are application modal. The frame of a modal window has a colored inner border; the frame of a modeless window does not.

8.2.2 Window Design Guidelines

Motif Only: The title of a dialog window includes the name of the application and describes the purpose of the dialog window. The window contains a separator between the control area and action area of the window.

Windows Only: The title of a message window is the application name and does not include the word "Error."

If a dialog window includes a text message, it uses language that is meaningful to users and requires no further documentation or translation. The text is left justified within the window. When a message contains more than one sentence, the important information is placed at the start of the message. The text is worded so that the action users are asked to perform can appear as a push button in the window. For example, a window displaying the message "Confirm deletion of file" contains Delete and Cancel push buttons. The application does not present timed-information windows (i.e., message windows that present information for a fixed time period) and then resume processing without requiring a user response.

³⁸ Expandable windows and Property and About dialog windows will be addressed in a future version of this document.

³⁹ Previous versions of Motif recommended that error, question, and warning message windows be application modal and information and working message windows be modeless. CDE recommends that all message windows be modeless whenever possible.

A dialog window contains at least one push button that either performs the dialog window action and dismisses it (e.g., OK) or dismisses the window without taking any action (e.g., Cancel). A default push button is available in each dialog window in the application. If an action executed in a dialog window results in an error that generates an error message window, the dialog window remains displayed while the error window is presented and then dismissed.

Push button order is OK/Cancel/Help in modal dialog windows, with OK designated as the default. Push button order is OK/Apply/Cancel/Help or OK/Apply/Reset/Cancel/Help in modeless windows; OK is the default in windows that perform single actions, and Apply the default in windows that perform multiple actions. <Cancel> has the same effect as selecting the Cancel push button in the window.

When a dialog window containing critical information is displayed, it is accompanied by auditory feedback (e.g., a beep) as a secondary indicator to attract the user's attention. Users can set auditory signals at a very low intensity or disable them as required (e.g., for rig-for-quiet operations on submarines).

8.2.3 Message Dialogs (Motif Only)

8.2.3.1 Error Message Windows

An error message window, shown in figure 8-5, is displayed to inform users when an error has occurred. The window contains the error symbol, a text message, and the following push buttons (in the order indicated):

OK, Help

Continue, Cancel, Help.

The text message describes the error, why it happened, and what should be done to correct it.

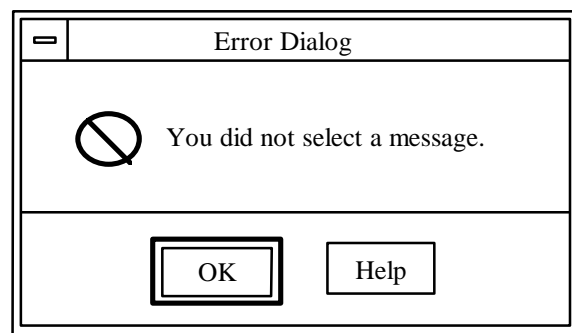


Figure 8-5. Example error message window in Motif.

8.2.3.2 Information Message Windows

An information message window, shown in figure 8-6, is displayed to convey noncritical information that requires acknowledgment by users (the message bar of a window is used for messages that require no acknowledgment). The window contains the information symbol, a text message, and the following push buttons (in the order indicated):

OK

OK, Help.

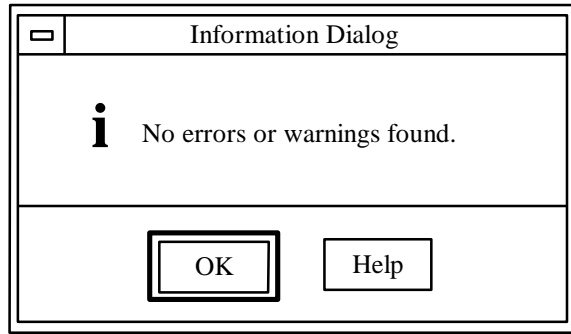


Figure 8-6. Example information message window in Motif.

8.2.3.3 Question Message Windows

A question message window, shown in figure 8-7, is displayed to request clarification of a previous response. The window contains the question symbol, a text message, and the following push buttons (in the order indicated):

Yes, No, Help.

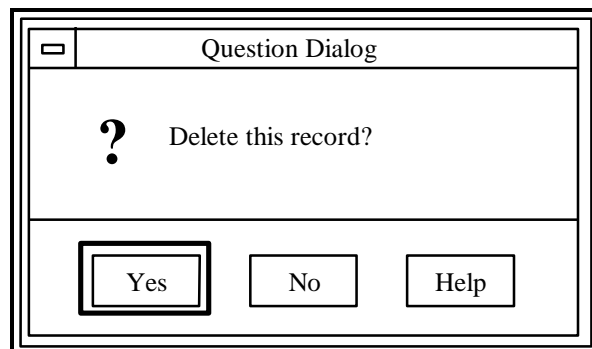


Figure 8-7. Example question message window in Motif.

8.2.3.4 Warning Message Windows

A warning message window, shown in figure 8-8, is displayed to present critical messages concerning the consequences of an action and to allow users to cancel a destructive action. The window contains the warning symbol, a text message, and the following push buttons (in the order indicated):

Yes, No, Help

Continue, Cancel, Help.

An audio signal accompanies the window to alert users to the warning.

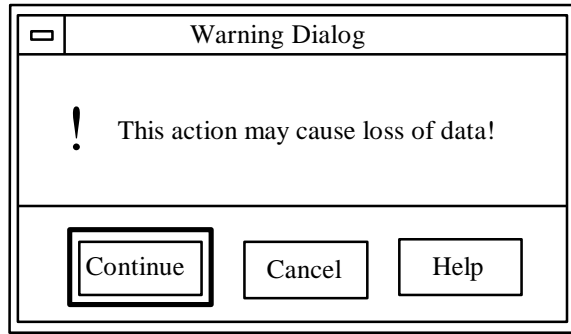


Figure 8-8. Example warning message window in Motif.

8.2.3.5 Working Message Windows

A working message window, shown in figure 8-9, is displayed when the processing time in response to a user's request exceeds 10 sec or when a user may want to cancel the operation that is in progress. The window contains the working symbol, a text message, and the following push buttons (in the order indicated):

- OK, Help
- OK, Cancel, Help
- OK, Stop, Help
- OK, Pause, Resume, Stop, Help.

Cancel interrupts the operation and returns the application and data to its state before the operation was activated. If a return to that state is not possible, Stop is used instead of Cancel. Stop interrupts the operation but does not reverse any changes already caused by the operation.

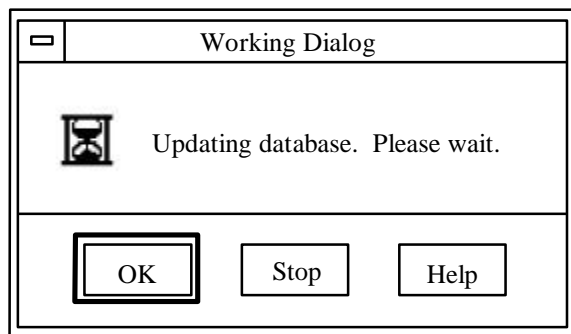


Figure 8-9. Example working message window in Motif.

During lengthy processing (in excess of one minute), a working window is updated (e.g., the text is changed or the trough in a gauge is filled) to indicate the status of processing if update information is available. The window remains displayed until the action is complete, the window doing the processing is minimized, or the user selects Cancel. When processing is complete, the window is removed (without user action). Users can cancel the operation in progress if desired, and they have to confirm the action before it is executed if unsaved data will be lost.

8.2.4 Message Dialogs (Windows Only)

8.2.4.1 Information Message Windows

An information message window, shown in figure 8-10, is displayed to provide information about the results of commands. The window contains the information symbol, a text message, and the following push buttons (in the order indicated):

OK

OK, Help.

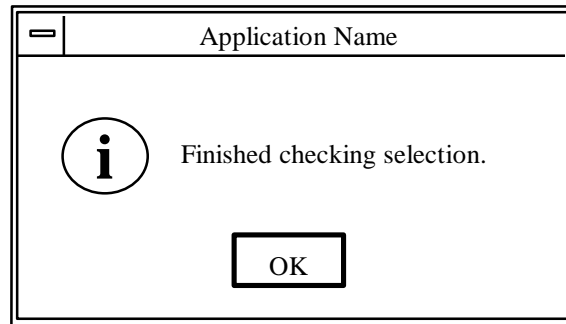


Figure 8-10. Example information message window in Windows.

8.2.4.2 Warning Message Windows

A warning message window, shown in figure 8-11, is displayed to present error information or to allow users to cancel a destructive action. The window contains the warning symbol, a text message, and a push button for each choice available in the window plus a Help button. The message text may be worded as a question; if so, the window includes the following push buttons (in the order indicated):

Yes, No, Help.

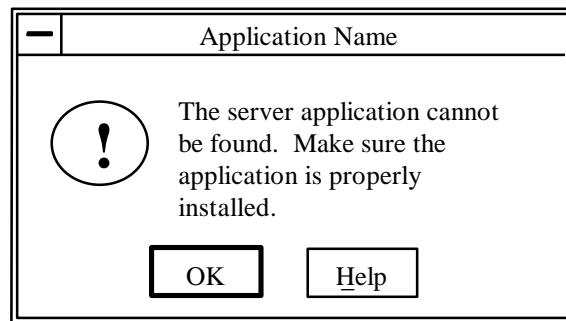


Figure 8-11. Example warning message window in Windows.

8.2.4.3 Critical Message Windows

A critical message window, shown in figure 8-12, is displayed to present messages that must be corrected before users can continue to work in the application. The window contains the critical symbol, a text message, and push buttons for each choice available in the window plus Help.

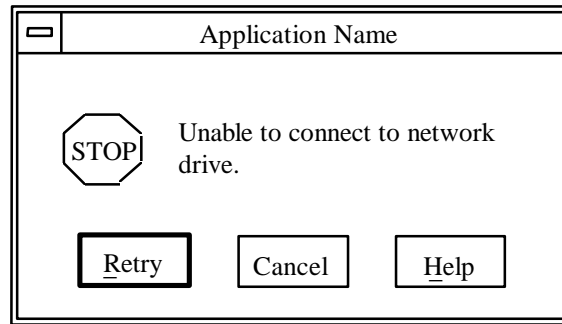


Figure 8-12. Example critical message window in Windows.

8.2.5 Selection Dialogs (Motif Only)

8.2.5.1 Command Windows

A command window is displayed when users need to enter keyboard commands. The window, shown in figure 8-13, contains a list box that displays a command history and a text field for entering commands; the window does not include any push buttons. The list has a vertical scroll bar when the command history exceeds the visible area in the list. The command history is cleared whenever the application is exited, and resumed when the application is launched again. The text field is wide enough for users to view and read an entire command; a horizontal scroll bar is not included unless command lines are unusually long.

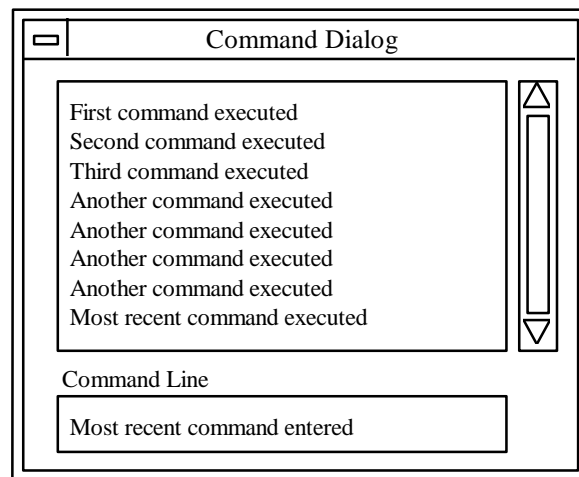


Figure 8-13. Example command window in Motif.

Selecting an item from the list displays it in the text field. <Enter> or <Return> executes the command and adds it to the bottom of the command history list. <Tab> moves the location cursor between the list and the text field. When focus is on the text field, <Up>, <Down>, <Ctrl><Home>, and <Ctrl><End> move the location cursor among items in the list and change the contents of the text field.

8.2.5.2 Prompt Windows

A prompt window, shown in figure 8-14, is displayed to request information needed to continue processing. A prompt window includes a message stating what information is needed, a text field for typing, and the following push buttons (in the order indicated):

OK, Cancel, Help

OK, Apply, Cancel, Help

OK, Apply, Reset, Cancel, Help

The text field has keyboard focus when the window is initially displayed.

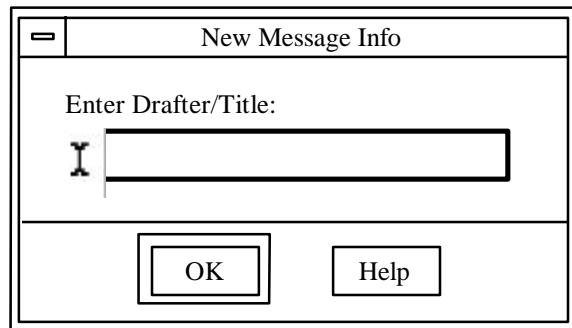


Figure 8-14. Example prompt window in Motif.

8.2.5.3 Selection Windows

A selection window, shown in figure 8-15, is displayed when users need to make a selection from a list of choices. The window includes a list box containing the choices available, a text field for displaying and editing the choice, and the following push buttons (in the order indicated):

OK, Cancel, Help

OK, Apply, Cancel, Help.

Both the list box and the text field include a heading that describes their contents. The list has a vertical scroll bar when the number of items exceeds the visible area in the list.

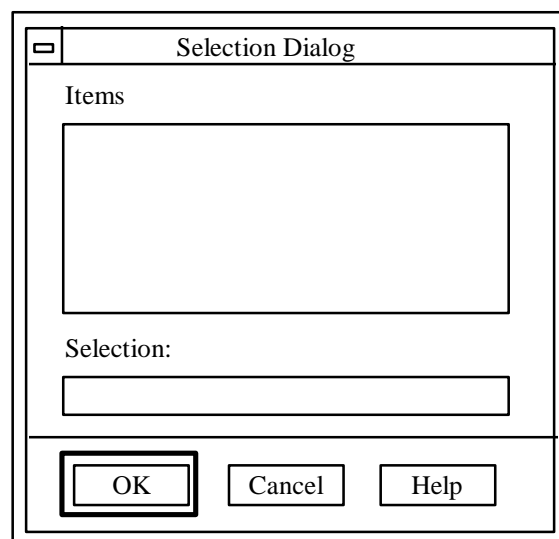


Figure 8-15. Example selection window in Motif.

Selecting an item from the list box displays it in the text field. If users type in the text field, the list scrolls to that item in the list. If the text typed in the text field does not match any items in the list, users are prompted to add the item to the list. When users select OK or press <Enter> or <Return>, the selection is executed and the window closed. <Tab> moves the location cursor between the list and the

text field. When focus is on the text field, <Up>, <Down>, <Ctrl><Home>, and <Ctrl><End> move the location cursor among items in the list and change the contents of the text field.

8.2.5.4 File Selection Windows

A file selection window, shown in figure 8-16, is displayed when users need to choose a file or directory.⁴⁰ A file selection window contains a text field for displaying and editing the current directory path, list boxes for displaying directory and file names, a text field for displaying and editing a file name, and OK/Update/Cancel/Help push buttons. The text fields and lists are labeled as in figure 8-16. The File text field can be omitted if the window is used to choose an existing file or directory. The OK push button can be replaced with a command that matches the action for which the window was displayed. When a file selection window is used to specify an existing file, the command is Open and it is the default action in the window. When the window is used to specify a new file name, the command is Save and it is the default action.

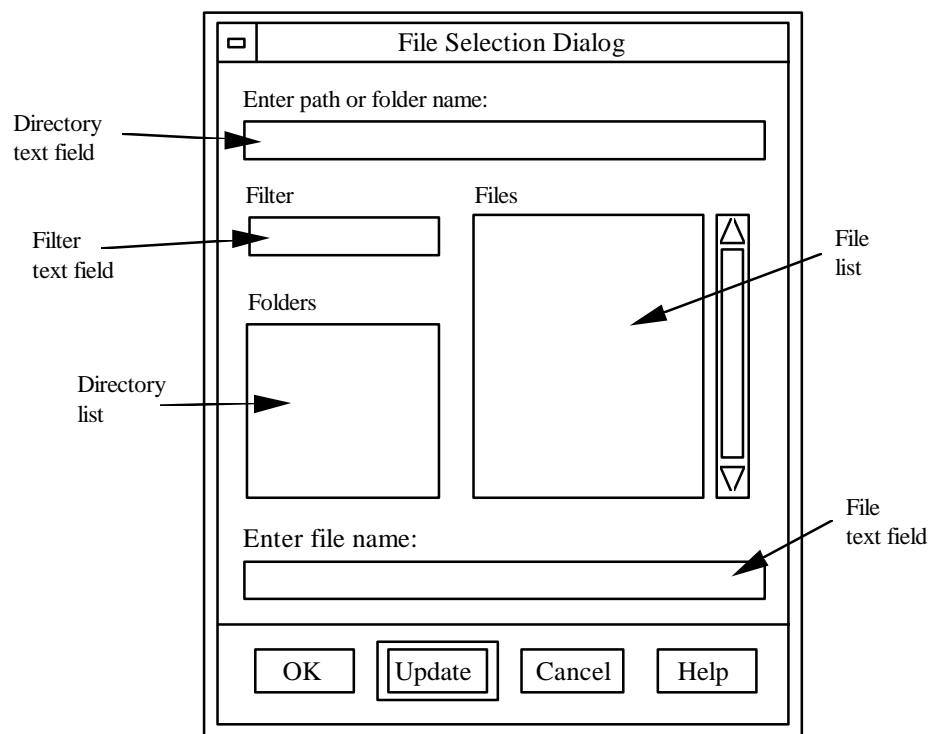


Figure 8-16. Example file selection window in Motif.

The items in the Directory and File lists are presented in alphabetical order. The first item in the Directory list is the parent directory and labeled “..”. The file selection window does not display hidden (i.e., dot) directories or files unless users need access to these types of files; if access is required, the window includes a check button that allows users to show or hide these files. The Directory text field presents the full path name; the File text field and the lists in the window show relative path names.

When users open the file selection window associated with a particular primary window, the directory location displayed is the default for that primary window. If users change the directory and then reopen the file selection window, the directory location is the one that was previously set by the

⁴⁰ CDE provides a file selection window with additional features beyond what was available in previous versions of Motif.

user. When users close the primary window, the directory location in the file selection window reverts to the default for the primary window. If the application supports multiple primary windows, the directory reverts to the default defined for that primary window. The default is selected to match the task being performed in the window. For example, when the user executes an action to save a file, the file selection window contains the user's home directory.

When users open a file selection window, the File text field has keyboard focus, and the File list displays the contents of the current directory. The File list is updated when users edit the Directory text field and press <Enter> or <Return> or when they select a directory in the File list. When users select a file from the File list, the file name appears in the File text field. The application executes the selection(s) in the window when users select an item in the File list and activate the OK (or comparable) push button, when users double click BSelect on an item in the File list, or when users select a file name and press <Return> or <Enter> when the File text field has keyboard focus. Users are prompted to confirm the action executed if it will overwrite an existing file.

8.2.5.5 Print Windows

A print window, shown in figure 8-17, is displayed when users need to select options for printing a file, a selection, or other type of object. The window contains a common area with standard information about the print job, an optional area with information specific to the application or function, and Print/Cancel/Help push buttons. The common area, located in the top part of the window, displays the name of the file or object type and includes controls for entering the name of the printer destination, the number of copies desired, and the text to appear on the banner page, if any. A text field or combo box can be used to enter the printer destination. The default entry in this field is "Default" (i.e., the printer that is the default destination in the system); users can select or type any other valid printer name. The application saves the last user entry in this field and displays it when the window is opened again. A text field or spin button can be used to enter the number of copies, and a text field can be used to enter the banner page title.

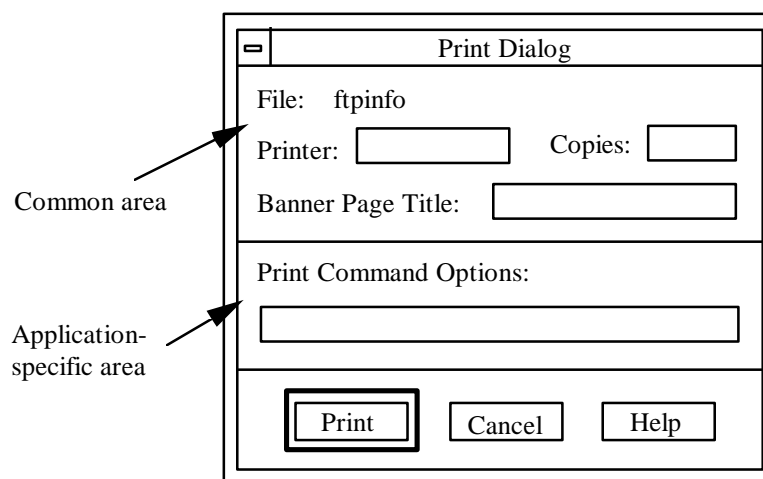


Figure 8-17. Example print window in Motif.

The same information is presented in the common area in all print dialogs, with any application-specific controls placed in the optional area in the lower part of the window. These controls request users to enter information such as Page Range, Priority, Orientation, Resolution, Paper Size, and Paper Source. If Reset and Print Preview actions are included in the window, they are available as push buttons and inserted between the Print and Cancel buttons. Separators are used between the common area, the optional area, and the push buttons.

8.2.6 Selection Dialogs (Windows Only) ⁴¹

8.2.6.1 File Open and File Save Windows

The standard File Open window is used to choose a file or directory. This window contains a noneditable text area displaying the current position in the directory tree, list boxes for displaying directory and file names, a text field for displaying and editing a file name, drop-down lists for selecting drives and file types, and OK and Cancel push buttons. Users navigate in the window by either selecting from the Drives or Directories control or by typing this information (along with a file name) in the File Name text field. When users switch to a new drive, the contents of the Directories list shows the contents of this drive, with the root directory at the top. Users can select a file by either typing in the File Name text field or selecting one of the items in the list box below the text field. If desired, users can filter the types of files included in this list by selecting a file type from the List Files of Type drop-down list.

A File Save As window has the same basic format as a File Open window, except the file names shown in the list box are grayed out (i.e., unavailable for selection) and the List Files of Type drop-down list is relabeled Save File as Type, with the type selected indicating the format of the file to be saved.

8.2.6.2 Print Windows

The standard Print window is used to submit a print job. This window identifies the printer to which the job will be sent and contains controls for specifying the print range, print quality, and number of copies and choosing to print to file or to collate the copies. These controls appear in the common part of the Print window, with application-specific controls available in an optional area in the lower part of the window.

8.3 CONSIDERATIONS IN WINDOW DESIGN

The specifications in this section are provided to assist developers in making effective design decisions regarding the windows in their applications. A key assumption underlying these specifications is that effective design begins with an understanding of the users' perspective and how the application will support the tasks they perform. Note: These specifications are not included in the UIS checklist.

8.3.1 Selecting Controls to Match User Actions

The controls that appear in a window match the actions that users are expected to execute in the window. The following are guidelines on how to choose the most appropriate controls:

Use radio buttons, an option menu, or a list box when users need to select from discrete values or choices; use a scale or spin button when users need to select from a continuous range of values.

Use a group of radio buttons when it is important for users to see all of the settings available in a group; use an option menu or drop-down list box when users need to see only the current choice.

Use radio buttons or an option menu when the set of options from which to choose is not likely to change; use a list box when the options from which to choose may change.

⁴¹ Other common selection dialogs in Windows will be addressed in a future version of this document.

When users have to make a single selection, use a group of radio buttons if there are up to 5-6 choices; use an option menu if there are up to 10-12 choices; and use a list box if there are more than 12 choices.

When users have to make multiple selections, use check buttons if there are up to 7 choices; use a list box otherwise.

Use push buttons for frequently executed actions when space is available to display the buttons; use tear-off menus for these actions if space is limited.

Use option menus for setting values or choosing from a set of related options; use push buttons for activating commands.

8.3.2 Arranging Controls by Importance and Scanning Order

The most important information and controls associated with the task being performed in a window is located in the upper left part of the window, working down to the less important information at the bottom of the window. In addition, information in a window is arranged to accommodate the possibility that users will resize the window. Placing the most important objects in the upper-left corner of the window allows users to manipulate these objects even if the window is reduced to a size that is smaller than normal. Finally, if a window contains task-critical information, it is visually set apart from other information in the window so that it can be easily seen. The minimum separation is one character space above and below and two character spaces before and after the critical information.

Windows are designed according to users' natural scanning order and probable selection sequences. In most cases, this order is from left to right and top to bottom. For example, when presenting a series of radio or check buttons, the most frequently used button is the top (or leftmost) button in the group. The window is designed from the perspective of what is logical to users and appropriate to the actions being executed, not what is logical or appealing to developers.

8.3.3 Designing for Efficiency in Task Performance

The controls in a window are arranged so that users can move quickly and easily among them. The amount of pointer movement and/or the number of keystrokes required to perform a task is minimized. Likewise, objects are arranged to minimize the amount of hand movement between the keyboard and pointing device when users work in a window; users are not forced to change input devices when performing a task.

The number of actions users have to execute in order to complete a task is minimized. For example, the application prefills text entry fields (e.g., current date, ownship name, position) with default data values whenever possible. Similarly, if the application repeats the same data fields in multiple windows, users have to fill the field once, with the application automatically placing this value in the field as the default whenever it recurs.

Developers need to adopt a consistent organizational scheme for the key elements in a window and then apply that basic scheme to all windows in the application. The same window design is employed whenever users have to perform the same basic task; for example, a single design is used to present identifying information on data records, whether the data are tracks or messages. Different or distinctive elements can appear in a window to fit the task being performed, but these elements are consistent across windows within the application.

The number of intermediate windows that users must interact with in order to complete a task is minimized. The application is designed so that users are able to perform each major task in a single window containing all of the information relevant to the task. Users can complete the task without having to refer to information not included in the window. Developers need to be sensitive to problems created by presenting users with a sequence of independent windows, each of which contains only a portion of the overall task being performed. While this type of design may be efficient from a software development perspective (e.g., by allowing re-use of previously designed windows), it can produce an unnecessarily complex task navigation problem for users, increase the difficulty of the task being performed, and place an unnecessary memory load on them to remember information not included in the current window.

8.3.4 Minimizing the Opportunity for User Error

The application is designed to minimize the opportunity for user error. Users can perform only legal operations, rather than allowed to execute an incorrect operation and then informed that they have made an error. Only those actions that are relevant are available for user selection. Controls that cannot be selected are both visually deemphasized by graying out to indicate their unavailability as well as disabled (so that no action is executed if they are selected by users). The application provides visual and behavioral cues to prevent users from performing an illegal or incorrect operation, rather than relying on error messages to inform users after an error has been made.

The actions available for each control displayed in a window match the action(s) users are expected to perform with the control. For example, if users are required to select a single item in a list box, the available selection methods are limited to this type of action. Users are not allowed to select multiple items in the list, after which they are informed in an error message that they should have selected only one item.

Efforts to minimize the opportunity for user error do not constrain users unnecessarily as they work in the application. For example, a window containing several mandatory data fields grays out the OK or Save push buttons until these fields are filled so that users are unable to save the data before completing the mandatory fields. This approach allows users to fill the fields in whatever order they choose or to make corrections as desired before attempting to save the data. A more constraining approach would manage each of the fields so that users have to enter a value before being allowed to leave the field. This approach forces users to perform data entry in a lock-step sequence defined by the application and restricts their ability to perform the task in a manner that does not match this sequence. The application needs to be designed to provide flexibility and support the user's sense of control, while at the same time bounding user interaction to minimize the opportunity for error.

9.0 INFORMATION PRESENTATION

9.1 TEXT INFORMATION

9.1.1 Text Font, Size, and Style

Text is presented using the default font, size, and style defined by Motif (i.e., CDE) or Windows (see section 7.1.6.2 and appendix D).

Motif: CDE distinguishes between system and user fonts, using a proportional-width font for the former and a fixed-width font for the latter. The system font is used for system areas such as menu bars, push buttons, toggle buttons, and labels, while the user font is used for text entered into windows. The Font control in Style Manager provides users with a choice of seven font sizes in which to display the system and user fonts.

Windows: Text in title bars and menus is presented in a 10-point sans serif bold font. Text in dialog windows and icon labels uses an 8-point sans serif bold font, while text in status bars is a 10-point sans serif, non-bold font.

When a choice of fonts is available to the application, a sans serif bold font is used so that text is readable when presented normally or grayed out, even if screen resolution is degraded.⁴² The font is of sufficient thickness and size to be readable when users are seated at a normal viewing distance from the screen. The DoD style guide indicates that at a minimum, character height should be 1/200 of viewing distance (e.g., a viewing distance of 36 inches requires a .18 inch character height on the screen).

9.1.2 Capitalization, Grammar, and Punctuation

Text (including titles and major headings) is presented in mixed case, following standard capitalization rules. All upper-case letters are used in text only for acronyms and abbreviations and for emphasis. Arabic rather than Roman numerals are used when information has to be numbered.

Continuous text (e.g., directions, messages to users) is phrased in simple sentences, in the affirmative (rather than negative) and in active (rather than passive) voice, as shown in figure 9-1. A sequence of events or steps is presented in the order they are performed. The referent for “it” or “they” in a sentence is easily identified. Normal punctuation rules are followed, and contractions and hyphenation are avoided. Paragraphs are kept short and separated by at least one blank line.

Do this:	Not this:
Press ENTER to continue.	The user should press ENTER to continue.
Clear screen before entering data.	Do not enter data before clearing the screen.
Select one.	Will you make a selection?

Figure 9-1. Example of wording style.

9.1.3 Acronyms and Abbreviations

Acronyms and abbreviations are used only when they are significantly shorter than the full word and are commonly understood by users (e.g., are related to normal language or are specific job-related terminology). Abbreviations are the shortest possible that will ensure uniqueness. Abbreviations are used consistently within the application. Words not commonly abbreviated are not abbreviated. The DoD style guide recommends that acronyms and abbreviations comply with the following documents:

AR 310-50. Authorized Abbreviations and Brevity Codes

MIL-STD-12D. Abbreviations for Use on Drawings, Specifications, Standards, and in Technical Documents

MIL-STD-411E. Aircrew Station Alerting Systems

⁴² Kobara recommends the use of sans serif, variable pitch fonts (e.g., Helvetica) for labels and system messages and serif, fixed pitch fonts (e.g., Courier) for text entry areas.

MIL-STD-783D. Legends for Use in Aircrew Stations and on Airborne Equipment

New acronyms are generated according to rules contained in MIL-STD-12D. When abbreviations or acronyms are used, a dictionary is available to users (e.g., as an option in the Help menu).

9.1.4 Formats for Date/Time and Latitude/Longitude⁴³

The application uses the following format when presenting date/time information:

The date is displayed as YYMMDD, where YY is the last two digits of the year, MM is the month, and DD is the day, or as DD MMM YY, where DD is the day, MMM is the month, and YY is the last two digits of the year.

Time is displayed as HHMM[SS]Z, where HH is the hour of a 24-hour day, MM is the minute, SS (optional) is the second, and Z is the time zone (Zulu [Z] time is the default).

Date/Time Group (DTG) is displayed as DDHHMMZ MMM YY, where DD is the day, HH is the hour, MM is the minute, Z is the time zone (Zulu is the default), MMM is the month, and YY is the year.

The application displays latitude and longitude in separate fields, with the labels “Lat” and “Long.” Latitude is displayed in one of the following formats:

D{D}H, where D (one or two characters) is the degrees of latitude and H is the hemisphere (N for North, S for South).

DD{MM{SS}}H, where DD is the degrees of latitude, MM is the minutes of latitude (optional), SS is the seconds of latitude (optional, but can only be given if minutes of latitude is given), and H is the hemisphere (N for North, S for South).

Longitude is displayed in one of the following formats:

D{D{D}}H, where D (one, two, or three characters) is the degrees of longitude and H is the hemisphere (E for East, W for West).

DDD{MM{SS}}H, where DDD is the degrees of longitude, MM is the minutes of longitude (optional), SS is the seconds of longitude (optional, but can only be given if minutes of longitude is given), and H is the hemisphere (E for East, W for West).

9.1.5 Wild Card Characters in Text Searches

Users can enter wild card characters to search for specific text patterns if this capability is appropriate to the functionality of the application.⁴⁴ The following wild card conventions are used if this search capability is available:

⁴³ The DoD style guide recommends these formats when presenting date/time and latitude/longitude information.

⁴⁴ The wild card characters included here are taken from the DoD style guide which recommends their use if appropriate to the application.

@ searches for the occurrence of a single upper- or lower-case alphabetic character. For example, abc@d retrieves the strings abcad, abced, and abczd; abc7d and abcd dd do not match the search pattern and are not retrieved.

searches for the occurrence of a single numeric character. For example, 123#4 retrieves the strings 12334, 12394, and 12304; 123x4 and 123554 do not match the search string and are not retrieved.

? searches for the occurrence of a single alphanumeric character (a - z, A - Z, 0 - 9, and punctuation marks). For example, abc?d retrieves the strings abcad, abcAd, abc(d, and abc9d; abcx d does not match the search string and is not retrieved.

* searches for the occurrence of zero or more alphanumeric characters. For example, abc*d retrieves the strings abcad, abcd, abfklsm d, and abc7d; abcd5 does not match the search string and is not retrieved.

9.1.6 Presenting Tabular Information

When information is presented in tabular form, each column of information has a heading, and the information in one column is clearly separated from that in other columns (usually by at least four character spaces). Data groupings are indicated with blank space, separator lines, and/or different intensity levels; multiple colors are used only if they provide additional meaning.

Alphabetic information is left-justified within a column, numeric information without decimals is right-justified, and numeric information with decimals is justified by the decimal point, as shown in figure 9-2. Long strings of numbers are delimited with spaces or commas to facilitate readability, and leading zeros are not used unless required for clarity. If the information extends beyond a single line, additional lines are indented to indicate they are continuations.

Do this:	Not this:
Artillery Tanks Jeeps Aircraft	Artillery Tanks Jeeps Aircraft
400 4210 38 3911	400 4210 38 3911
1.5 10.38 1.365 500.0	1.5 10.38 1.365 500.0

Figure 9-2. Example of data justification within columns.

Tabular information is grouped or arranged so that users can identify similarities, differences, trends, or relationships. For example, depending on the purpose of the window, the information can be presented in sequential, spatial, alphabetical, functional, or chronological order. Information that is

particularly important, requires immediate user response, and/or is used more frequently is presented first in the table.

9.2 GRAPHICAL INFORMATION

9.2.1 Line Graphs and Surface Charts

Line graphs, such as those shown in figure 9-3, are used to present trend information, spatially structured information, time critical information, or relatively imprecise information. The axes of the graph are clearly labeled and include the unit of measurement as appropriate. The labels are in mixed case and oriented left to right (including the vertical axis of a graph) for normal reading. The minimum and maximum value are indicated on each axis, with up to nine intermediate markings showing gradations on the axis. The starting point on each axis is zero, with the gradations indicated in whole numbers, unless a zero starting point is inappropriate for the data being displayed. The gradations are at standard intervals (e.g., 1, 2, 5, 10), with intervening gradations consistent with the labeled scale interval.

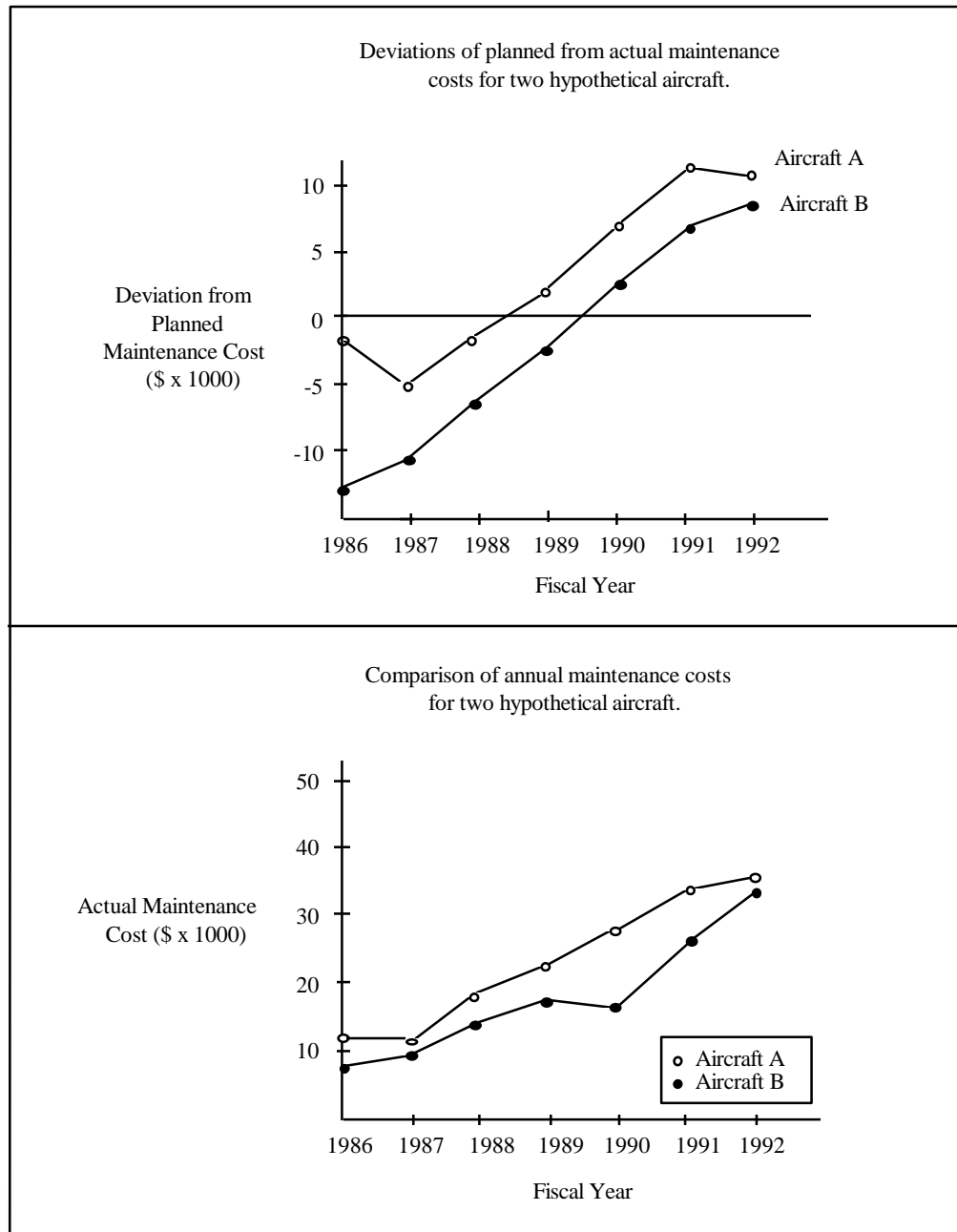


Figure 9-3. Example line graphs.

Labels are used instead of legends or keys when it is necessary to identify the data that are plotted on the graph. The labels are oriented horizontally and located next to the data being referenced. Each line or curve on a graph is labeled and coded (e.g., with solid, dashed, dotted lines), and a line or curve containing critical or abnormal data is coded (e.g., by color, line thickness, annotation) to call attention to that part of the display. If grid lines are included in a graph, they are unobtrusive and do not obscure the data presented in the graph; users can display or suppress grid lines as desired.

A line graph is limited no more than five lines/curves, with each one identified by an adjacent label (rather than in a separate legend). If corresponding data are presented in multiple graphs, the same

coding scheme is used in each graph. Coding is also used to highlight more important or critical information or to identify actual from projected data within the graph.

If users are required to compare multiple trend lines, the lines are presented on a single graph. If the lines have to be presented in multiple graphs, users can redraw the graphs using the same scale on both graphs to facilitate comparison. If users have to read precise values from a graph, options are available to display the actual data values on the graph and to zoom the display if necessary. In addition, aids are provided for scale interpretation (e.g., displaying a grid upon request, providing vertical and horizontal rules that the user can move to the intersection point, or letting users click on a point on the graph and having the exact values displayed in a pop-up window).

A surface chart is a type of line graph in which the data being depicted represent all parts of a whole. The curves/lines are stacked above one another to indicate aggregated amounts, and the area between each curve/line is coded using different colors, shadings, or textures and identified by a text label displayed within the area. If a surface chart is used, the data categories are ordered to reflect the logical organization of the entity being displayed. If no a priori organization exists, the data categories are ordered so that the least variable ones are at the bottom and the most variable at the top.

9.2.2 Bar Charts and Histograms

A bar chart, such as the ones shown in figure 9-4, is used to compare a single measure at several intervals, and a histogram to compare a single measure when the number of intervals is large. The bars in a related set of bar charts have a consistent orientation (vertical or horizontal), and bars containing data that must be compared are presented adjacent to one another. Frequency counts are usually displayed in vertical bars, and time durations in horizontal bars. If the displayed data have to be compared with a critical value, a reference index is provided.

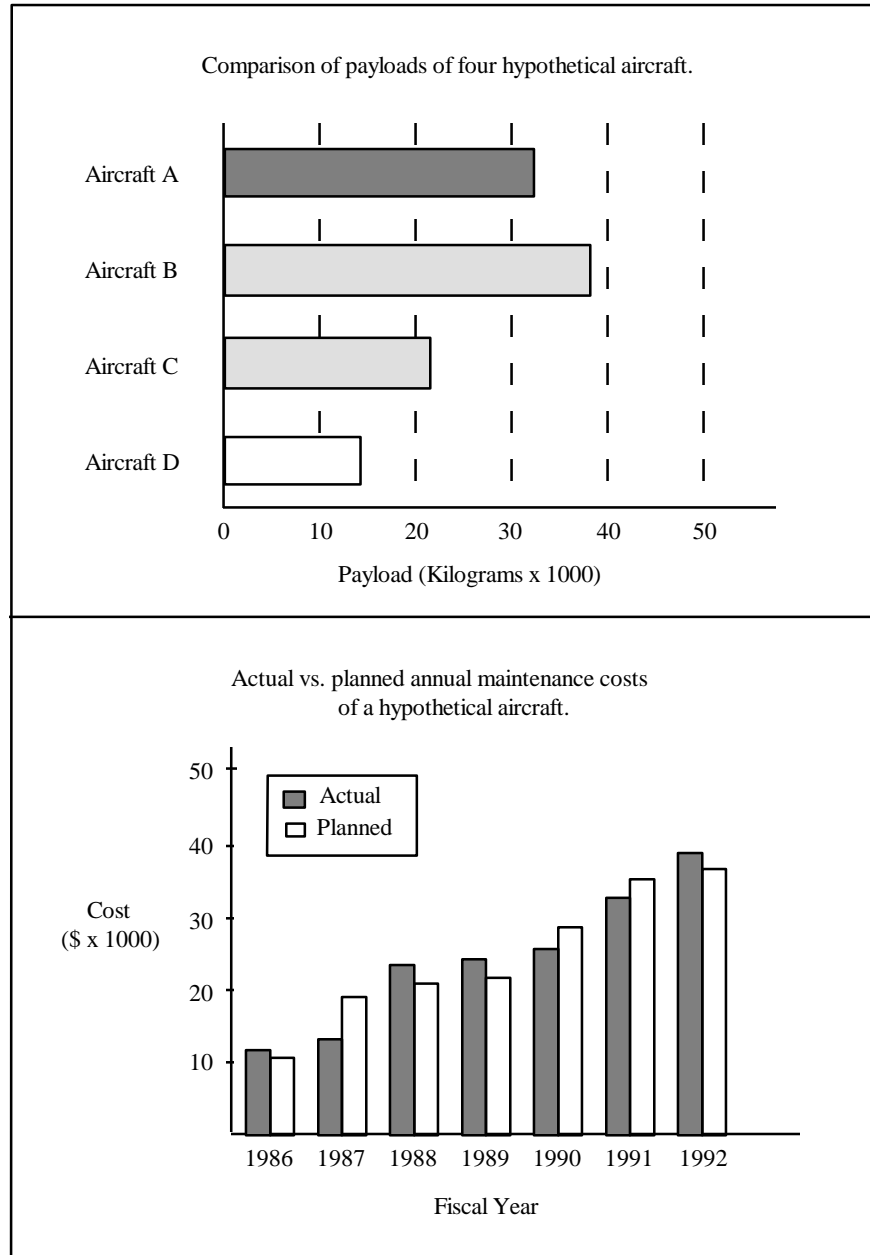


Figure 9-4. Example bar charts.

If the number of bars being displayed is small, a bar chart is used, with the bars separated, using one-half or less of the bar width as the spacing between bars. If the number of bars is large, a histogram is used (i.e., eliminating the spacing between the bars). Coding (e.g., color, shading, texture) is used to distinguish among different groups of bars or to highlight important data in one or more of the bars. If multiple bar charts or histograms are presented, related groups of bars are presented in a consistent order in each one. Each bar is identified with its own text label, rather than presenting the labels in a separate legend.

The bar chart or histogram is designed to conform to user expectations. Charts and axes are clearly labeled, and important information is highlighted. When bars are presented in pairs, they are labeled as a unit, with a legend provided that distinguishes between the bars.

Stacked bars are used when both the total measures and the portions represented by segments are of interest. This arrangement of bars is similar to a surface chart. If stacked bars are used, the data categories are presented in the same sequence. As with surface charts, data categories are ordered so that the least variable are at the bottom of the bar and the most variable are at the top. The areas within each bar are coded using different colors, shading, or texture and identified by a text label displayed within the area.

9.2.3 Flow Charts

A flow chart is used to present a schematic representation of sequences or processes. The path indicated in the flow chart is left to right, top to bottom, or clockwise. Each decision point in the flow chart contains a single, simple decision, as shown in figure 9-5. The elements and lines are coded (e.g., symbol and shape coding) to assist in understanding, and the same coding scheme is used throughout the flow chart. For example, the flow chart provides directional indicators (arrows) to indicate the sequence to be followed. A legend is included that describes each element and code used in the flow chart, and critical information and/or steps are highlighted.

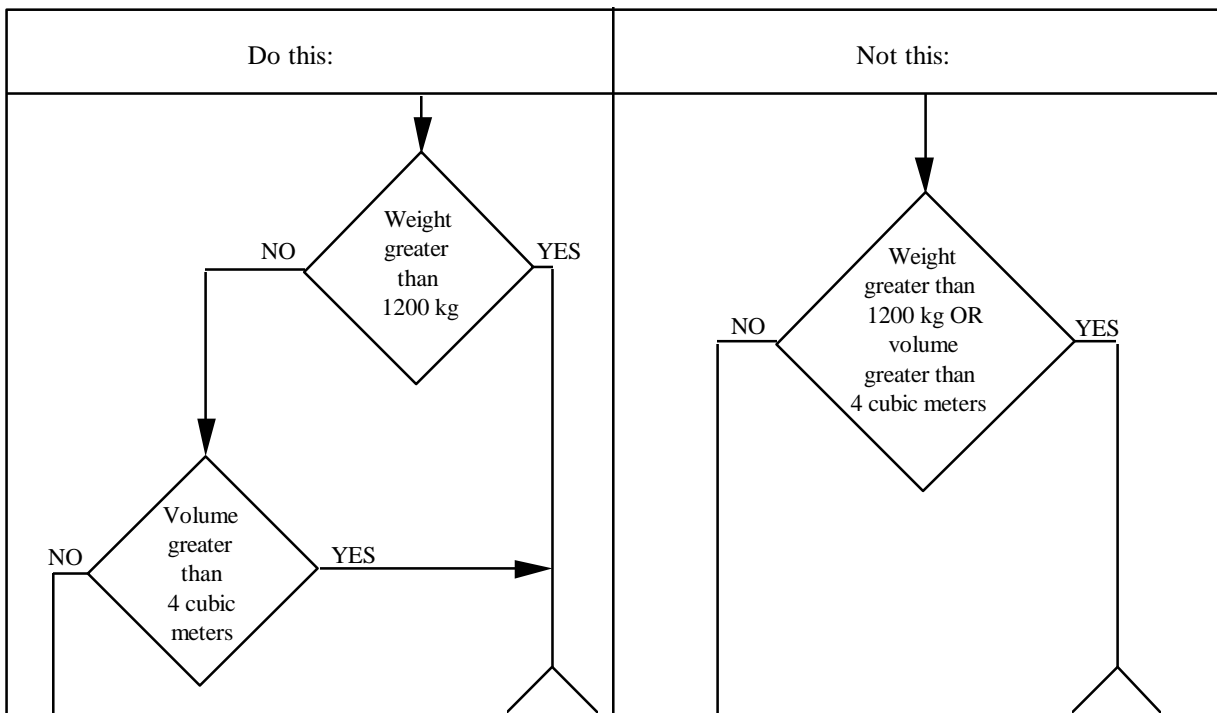


Figure 9-5. Example of labeling decision points in a flow chart.

The steps presented in a flow chart are ordered logically (i.e., follow the sequence of operations, steps, or processes from start to finish), or the most important decisions or the decisions that can be made with the greatest certainty are placed first. If no ordering scheme can be identified, the flow chart is organized to minimize the length of the path through it.

The shapes (e.g., boxes) used in the flow chart follow existing shape coding conventions, and the text presented in the chart is oriented for normal reading. Important elements (e.g., paths through the chart) are emphasized through coding such as color.

9.2.4 Pie Charts

A pie chart (shown in figure 9-6) is used to provide an approximation of how an entity is apportioned into component parts. If an accurate estimate of proportions or quantitative information is needed, a bar chart is used instead of a pie chart.

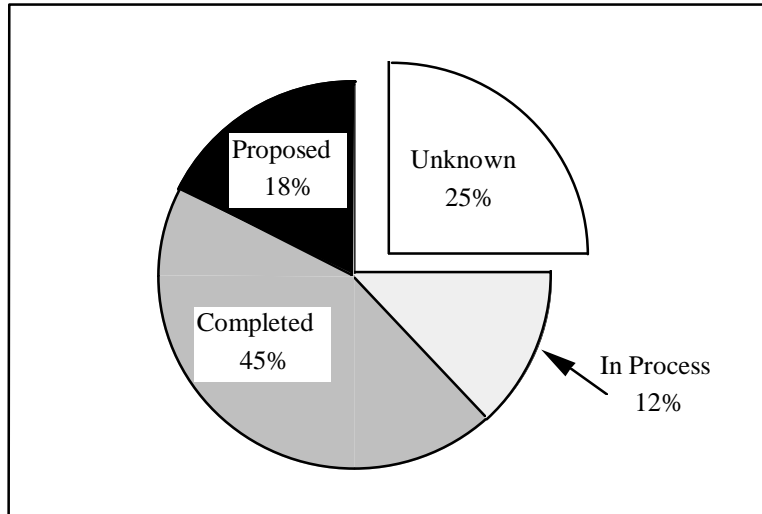


Figure 9-6. Example pie chart.

Each segment in a pie chart is coded using different colors, shadings, or textures and identified by a text label (presented in normal orientation) within the area. If the segment is too small to contain the label, it is placed outside the segment, with a line from it to the segment. The label describes the content of the segment and includes the number (i.e., percentage or actual value) being represented by the segment. Segments are emphasized by using special shading and by displacing them slightly from the remainder of the pie chart.

9.3 TACTICAL INFORMATION

9.3.1 Tactical Symbolology

Tactical symbolology and graphics conform with available military standards. MIL-STD 2525 defines a common warfighting symbolology for command and control systems fielded by the military services.⁴⁵ The DoD style guide recommends that map graphic symbols conform with published standards such as NATO Standardization Agreement 2019 Military Symbols for Land-Based Systems, Army Field Manual 101-5-1 Operational Terms and Symbols, and the DIA Standard Military Graphics Symbols Manual. Applications with specific operational domain requirements are to comply with all relevant national and international symbolology standards (e.g., Electronic Chart Display and Information Display System requirements for navigation systems published by the International Hydrographic Organization). If the application creates new symbolology, it does so in ways that are consistent with applicable standards (e.g., pointed shapes are “hostile,” round or curved shapes are “friendly”) and match user expectations.

⁴⁵ Implementation of the symbolology is recommended in version 1 of the document but will be mandatory when version 2 is published in 1996.

9.3.2 Coding of Tactical Information

If color is used to impart tactical meaning, it is used as a redundant code with another display feature (e.g., size, shape, text) and is not as the sole basis for coding. When color coding is used, each color represents one category of tactical data. If the application uses color to indicate threat status, it does so in accordance with applicable military standards. If the application uses color to indicate system status, it adheres to the following conventions:

Green/Blue = Operational/Normal/Noncritical
Yellow = Caution/Questionable
Red = Inoperative/Error

The background color behind text is not changed to show a change in system status because changing the background color usually reduces the readability of the text. Instead, the change is signaled by changing the color of an object (e.g., a box or circle) next to the text. The colors selected to convey tactical meaning are the same throughout the application, and their use is restricted to that function only. If one of these colors is assigned another meaning, a different shade is selected so as to minimize the likelihood of confusion with the convention.

Color can also be applied to tactical information for the purpose of alerting. In this case, only the information to which the application wants to direct user attention is assigned a unique color. As in color coding, a standard meaning in terms of alert criticality is assigned to each color, and that color is used only to convey this meaning. While alerting is usually indicated by assigning color to text information (e.g., in a list or table), colored icons can also be defined and appended to the information. Examples of how color might be applied as an alert indicator are provided below:

Vulnerability Time

Red = Vulnerable now
Yellow = Vulnerable in X minutes
Green = Vulnerable in Y minutes
(where Y is greater than X)
Blue = Not vulnerable

Confidence Factor

Red = Unknown
Yellow = Low
Green = High

Probability of Hostile Action

Red = Imminent
Yellow = Probable
Green = Possible
Blue = None

Probability of Detection

Red = High probability (greater than X percent)
Yellow = Medium probability (less than
X percent but greater than Y percent)
Green = Low probability (less than Y percent)

Priority

Red = High
Yellow = Medium
Green = Low

Action Items

Red = Now
Yellow = In X minutes
Green = No time limit
Blue = No action required

9.4 INFORMATION CODING

9.4.1 Color

Color is used redundantly and only to provide required functionality; coding methods other than color are applied whenever possible. The addition of color can increase response time and the likelihood of error due to color confusions. Color is a more effective code for search tasks and symbol identification tasks than other cues such as shape, size, or brightness. The performance advantage of color coding increases with the density of the symbols in a display and when the number of nontarget

symbols of a different color than the target increases. However, color used excessively or inappropriately in displays can degrade user performance when compared to monochromatic displays.

The number of colors used to code the information in an alphanumeric display does not exceed seven, and only four codes are displayed at any one time. The number of colors used to code information on graphical displays does not exceed eight or nine, since users have difficulty discriminating among more than this number of colors. When information in a display is color coded, users have the option of displaying the meaning of the code as a reminder (e.g., in the message bar). Slight shade changes in color are not used to show gradation or choice. Shade differences are usually difficult to see, especially on a varied background such as a map. While normally discouraged as a graphical area discrimination technique, color shading, if used, is of sufficiently differing intensity as to be obvious and is not used to determine object selection or for control of the application.

Mayhew in Principles and Guidelines in Software User Interface Design provides the following ISO guidelines regarding color coding:

<u>To denote:</u>	<u>Use:</u>
Larger size	Saturated or bright colors
Smaller size	Desaturated or dark colors
Equal size	Colors equal in brightness
Heaviness	Saturated, dark colors
Lightness	Desaturated, light colors
Depth	Saturated, dark colors
Closeness	Saturated, bright colors
Height	Desaturated, light colors
Low-end continuum	Short-wavelength dark colors
High-end continuum	Long-wavelength bright colors

The DoD style guide provides the following additional guidelines on using color in computer display systems:⁴⁶

- a. Highly saturated colors, opposing colors (e.g., yellow and blue), and colors at spectral extremes (e.g., yellow and purple) are not used together because they may cause afterimages, shadows, and depth effects.
- b. Pure white text is not displayed on a pure black background because this combination produces halation which makes the text less readable. Saturated blue is used only for background features in a display and not for critical data or for small lines or dots when the background is dark.
- c. Both brightness and type of lighting (e.g., incandescent vs. fluorescent) can affect how colors are perceived. For example, bright ambient light desaturates display colors, leading to degraded color identification and discrimination.
- d. At normal viewing distance from a screen, maximal color sensitivity is not reached until the size of a colored area exceeds about a three-inch square. Smaller size images become desaturated and change slightly in color. Also, small differences in actual color may not be discernible, and small adjacent colored images may be perceived to merge or mix.

⁴⁶ More detailed guidelines on color use can be found in the DoD style guide.

e. Color discrimination is better when color images are displayed on an achromatic background (black, gray, or white) and achromatic images are displayed on a color background. If color images are displayed on color backgrounds, then background and symbol colors should contrast in both brightness and hue to ensure legibility.

9.4.2 Flashing

Flash coding is used only to display urgent information for user attention. No more than two levels of coding are used. The flash rate is in the range of 3-5 Hz with equal on/off intervals; if two levels of flashing are used, the second flashes at 1-2 Hz, with equal on/of times. When flash coding is applied to a displayed item, a flashing symbol (such as asterisks) is used rather than flashing the text itself. Users are able to acknowledge the event causing the flashing and suppress it if desired.

Windows Only: The application can flash the title bar of a window to draw attention to its contents. The flashing is accompanied by an auditory signal (e.g., one or two beeps) as a redundant cue in the event the flashing is not visible (e.g., the window is obscured).

9.4.3 Reverse Video

Reverse video is not used for coding since it is used in Motif and Windows for highlighting (e.g., to indicate that an object has been selected). Also, although effective in making data stand out, reverse video can reduce legibility and increase eye fatigue.

9.4.4 Size and Shape

If size coding is used, the number of size codes is limited to five or less, and users are required to interpret relative size rather than absolute size. Care should be taken when using size and color coding together since users' perception of object size can be manipulated by varying the color saturation and lightness of the object.

If shape coding is used, the number of shape codes is limited to 10-20, and the shapes used relate to the object or operation being represented. The color and detail added to the shapes are the minimum needed for users to identify the meaning assigned to the shape.

9.4.5 Sound

Auditory signals are used to alert users to critical conditions or operations. If auditory signals are associated with noncritical operations (e.g., as an alternate means of information presentation), they are used sparingly and users can acknowledge and turn off the signal at their discretion. Auditory signals are intermittent in nature and allow sufficient time to respond; they are distinctive in intensity and pitch, and the number of signals provided to users does not exceed four. The intensity, duration, and source location of the signal are selected to be compatible with the acoustic environment of users and the requirements of other personnel in the area surrounding the system.

9.4.6 Text Font and Style

If text font and/or style is used for coding, no more than two styles of type (e.g. regular and italics) or two weights (e.g., regular and bold) are available at one time. In addition, variations in type size are limited to no more than three at any one time. Capitalization can be used for emphasis in text but is not the sole indication of critical information in a window. While underlining is also effective in drawing user attention to specific text information, it can reduce legibility and so is used sparingly in the application. In addition, in hypermedia software, the underlining of a word or phrase is used to

indicate the presence of a link to other information. If underlining is used in the application, it is implemented in ways that do not conflict with this convention.

9.5 DYNAMIC INFORMATION

Users can control the rate at which dynamically changing information is updated. In addition, they can freeze the display of any information that is being updated automatically and resume the updating either at the point of stoppage or at the current point in time. When users have to read dynamically changing information reliably and accurately, the update rate is no more than once per second. When users have to identify the rate of change or read gross values, the update rate is 2-5 times per second. Users are prompted to return to automatic updating after freezing a dynamic window (e.g., while users execute a print command) and are informed if significant changes in data occurred while the display was frozen.

Similar capabilities are available to users when interacting with auditory information. Users can control the playback of auditory information (e.g., start, stop, pause) as well as adjust the volume of the playback.

10.0 TASK-SPECIFIC WINDOW DESIGN

10.1 DATA ENTRY WINDOWS

A data entry window, shown in figure 10-1, provides a template to display, enter, change, and delete data. The data fields in the window are organized by sequence of use, frequency of use, or importance, with related fields appearing together and separated from unrelated fields. If users are entering data from a hardcopy form, the window format is identical to the hardcopy format. When a data entry window contains different kinds of controls for data entry (e.g., text fields, option menus), the controls are arranged for efficient data entry using the pointing device or keyboard and to minimize hand movement between input devices.

Current Position/History					
Track:			Name:		
Type:			Hull:		
UIC:			Class:		
DTG	Lat	Long	Sensor	Course	Speed

Figure 10-1. Example data entry window in Motif.

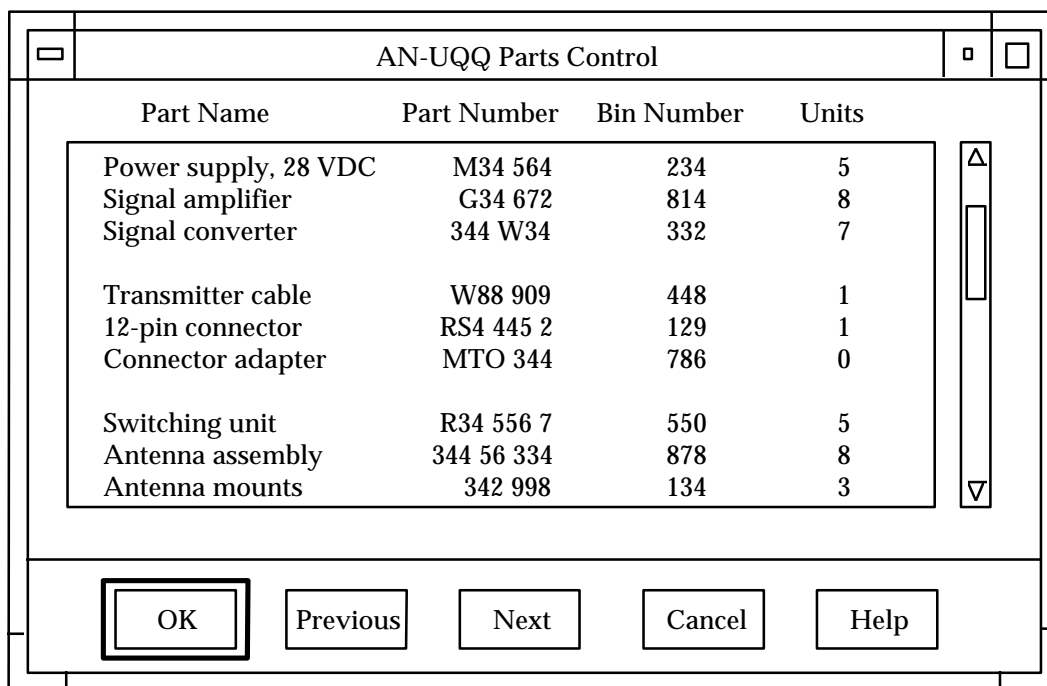
If the data being entered are tabular, the window arranges the data entry areas in rows and columns, with each one labeled; users are not presented with an empty text window, with no clues as to

format, for entering the information. If the data entry area is scrollable, the row and column headings are placed along the edges of the window and remain visible when the window is scrolled. In a group of related data fields, the labels and text fields are both left justified, or the labels are right justified and the text field left justified (as in figure 10-1). A conditional (or dependent) field is placed to the right of or below the field to which it relates. The field can be either unavailable (i.e., grayed out) or not displayed at all until the control to which it relates is selected.

Users can obtain information about a data field and its contents. Options available for providing this information are to automatically display it in the message bar of the window when the field has keyboard focus or to include it in the Help window that accompanies the data entry window.

10.2 TABULAR DATA WINDOWS

A tabular data window, shown in figure 10-2, is used to organize and display alphanumeric information in tabular or columnar form. The window includes vertical and horizontal scroll bars if the information being presented exceeds the space available in the window. If the information can be scrolled horizontally, the column heading scrolls with its associated column. If the information can be scrolled vertically, the column heading is placed outside the scrollable area and remains visible when the column data are scrolled, and the window includes controls for paging (e.g., Next and Previous push buttons). When users page through the information, the last line on one page is the first line on the next page. Whenever possible, the content of the window is arranged so that it does not extend over more than one page horizontally.



Part Name	Part Number	Bin Number	Units
Power supply, 28 VDC	M34 564	234	5
Signal amplifier	G34 672	814	8
Signal converter	344 W34	332	7
Transmitter cable	W88 909	448	1
12-pin connector	RS4 445 2	129	1
Connector adapter	MTO 344	786	0
Switching unit	R34 556 7	550	5
Antenna assembly	344 56 334	878	8
Antenna mounts	342 998	134	3

Figure 10-2. Example tabular display window in Motif.

A tabular data window window can be used for display only (as in figure 10-2), or controls can be provided that allow users to manipulate the information. In figure 10-3, the information in the window is presented in a multi-column list, with the items in the list forming the individual records that appear in the rows. Users are able to sort the records that appear in this type of matrix. The headings that can be sorted appear as buttons so that clicking on the heading sorts the records in an order based on the items in that column. The heading remains highlighted after being selected to indicate the column that

was sorted. If additional sort variations are needed, they are provided in menus or push buttons in the window. If desired, a speed search capability can also be included in the window so that users can both sort the items in the list and then execute a speed search to scroll to the first instance of an item that begins with a particular letter.

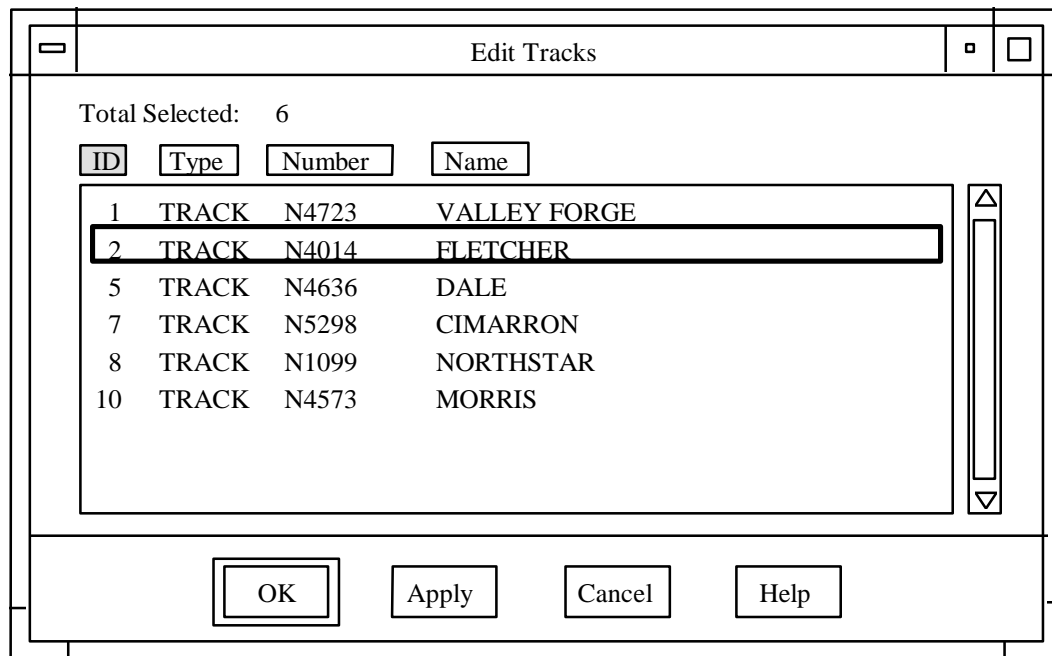


Figure 10-3. Example tabular display window with sort capability in Motif.

10.3 LIST-TO-LIST TRANSFER WINDOWS

A list-to-list transfer window, shown in figure 10-4, is used to move objects from one collection to another. This window contains a source list on the left and a destination list on the right, separated by two push buttons that allow items to be transferred between the two lists. The push buttons can contain text labels (e.g., Add, Remove) or left and right arrows indicating the direction of the transfer. When keyboard focus is on the source list (and one or more of the items is selected), Add is displayed as the default and Remove is disabled. Conversely, when focus is on the destination list, Add is disabled and Remove is shown as the default. The window can include radio and check buttons or option menus that allow users to modify the contents of the source list (e.g., to limit the items in the source list to those with specific features).

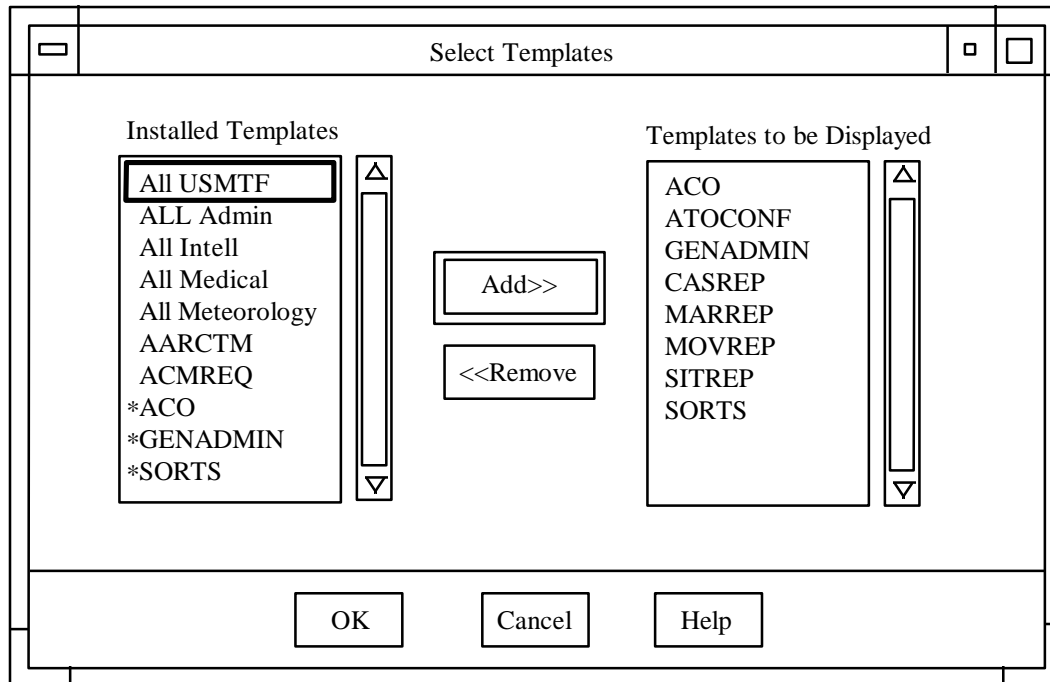


Figure 10-4. Example list-to-list transfer window in Motif.

Users can transfer more than one item at a time between lists but cannot transfer multiple instances of the same item to the destination list. Depending on the nature of the transfer task, an item in the source list can be either copied or moved when users transfer it to the destination list. In the former case, the item is marked (e.g., with an asterisk) to indicate that it has been transferred. The mark is removed when users transfer the item back to the source list.

10.4 MAP WINDOWS

10.4.1 Map Information

A map window, shown in figure 10-5, includes identifying information about the map (e.g., map name, coordinates, area, scale) along with status information (e.g., “drawing map”). This information is presented in the message bar of the window or in a subarea of the window itself. A continuous coordinate indicator giving the pointer location on the map is available in a standard part of the window (e.g., with the identifying information at the bottom of the map).

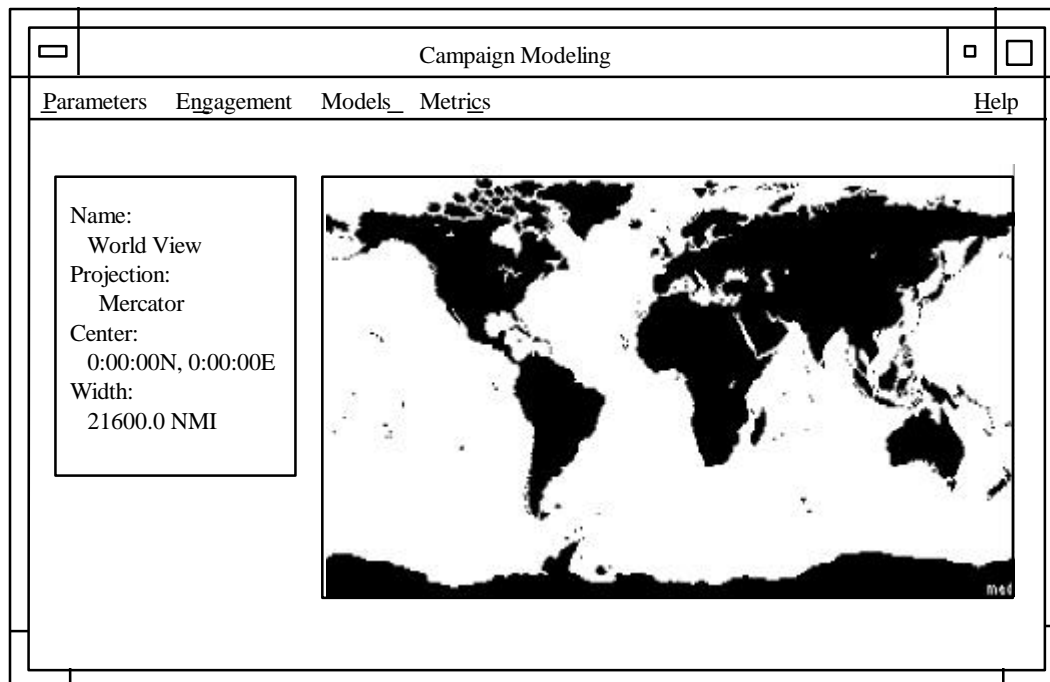


Figure 10-5. Example map window in Motif.

Each map is displayed using the same orientation (i.e., north toward the top of the window), and the important features are labeled. The labels are positioned consistently with respect to the feature they describe (e.g., next to or below the feature), do not obscure important information or clutter the map, and remain legible at all map resolutions. Users can select map features and define their appearance when they are displayed (e.g., select country colors).

Controls for map manipulation are displayed directly in the window or available in individual dialog windows. Users can pan and zoom the map as desired. A position or change indicator is provided as a means for users to return quickly to the normal or starting map. In addition, users can define a baseline (i.e., home) position on a map and return to this position quickly (e.g., by selecting the appropriate menu item).

Users can determine the distance and bearing between any two points on the map. Other functions available to users include areal computation/verification, area of interest selection, and area bounding boxes. Users can enter latitude and longitude to the level of accuracy needed (e.g., degrees, minutes, seconds, tenths or hundredths of seconds); when calculations such as range, bearing, and position are performed, the answer computed reflects the degree of accuracy appropriate to the scale of the map displayed.

10.4.2 Map Objects

Map objects such as symbology are placed on the map accurately or connected to the desired location using arrows, lines, or other graphics. The label for an object appears next to the object and presents essential information (e.g., unit or track identification) about it. The background of the object and label is transparent so as not obscure other information (such as overlays) displayed on the map. The intensity of the map is adjustable so that selected portions of the map can be faded out without losing all map features. If multiple sets of map objects are available, users can select the set desired and switch between sets without losing data.

Users can add, edit, reposition, and delete map objects such as tactical graphics and overlays and change the appearance of information about these objects on the map. When a map is zoomed, the size of these objects (including labels) is adjusted so that users can read them. Objects such as overlays include visual indications (i.e., “handles” displayed on the object) to define the portion of the graphic that is selectable and show where the graphic will be positioned when it is moved to a new location.

The pointing device selection methods listed in table 10-1 are used to select and deselect objects in a map window. If additional selection methods are implemented, they conform to those presented here and in table 3-1; e.g., the pointer indicates the locus where pointing device operations occur. Keyboard methods for selecting map objects conform to methods presented in tables 3-2 or 3-3.

Table 10-1. Pointing device selection methods in a map window.

Single Selection:

To select one or more map objects one at a time, position the pointer on an object (i.e., the graphic or its label) and click BSelect. If previously unselected, the object is selected; any objects that were selected remain selected. Add objects to the selection by clicking on other unselected objects.

Range Selection:

To perform a range selection, position the pointer near the first object in the range, then press BSelect to set the anchor for the range. Drag the pointer until it is beyond the last object in the range, and release the button to complete the selection. As the pointer is dragged over the objects, a bounding box is displayed outlining the objects being selected. When BSelect is released, the box disappears and the objects that were in the box are selected.

Deselection:

To deselect a single object, position the pointer on an object and press <Shift> and click BSelect; the object is deselected and returns to its normal appearance.

To deselect all objects, position the pointer on an empty part of the map, and double click BSelect; all previously selected objects are deselected and return to their normal appearance.

Users can view or declutter overlapping map objects and obtain additional information, including exact map coordinates, for selected objects (e.g., by double clicking on the symbol). Users can distinguish among objects that represent coincident points and obtain information that will allow them to resolve ambiguities among the objects. Users can make selections quickly and accurately in collections of densely packed or closely overlapping objects (e.g., by selecting from a pop-up menu listing all of the objects).

When users display an object such as a color overlay on a map, a color coding key is also displayed (e.g., in a subarea of the map window or in a separate dialog window) so that users can interpret the information in the overlay. Users can display the coding key as desired without having to redisplay the overlay. If the coding key is presented in a dialog window, the window is the minimum size needed to present the required information and positioned so that it obscures as little of the information on the map as possible. If appropriate, the key functions as a scale so that users can interpret the coding in the overlay easily and accurately.

10.5 GRAPHICAL SCHEDULING WINDOWS

10.5.1 Schedule Design

A graphical scheduling window is used to display timelines or scheduled events. The schedule is displayed with time presented on the horizontal axis and the tasks to be performed arrayed vertically, as shown in figure 10-6.

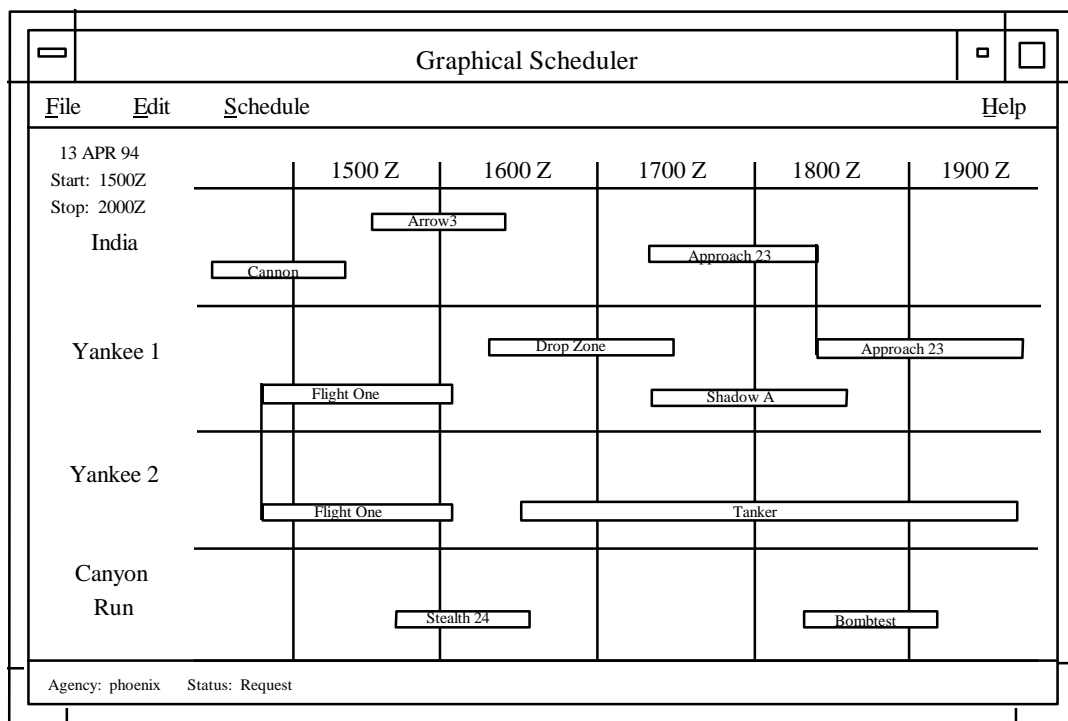


Figure 10-6. Example graphical scheduling window in Motif.

Each event in a schedule is represented by an event icon (in the form of a line or bar) whose length is proportional to the amount of time necessary to complete a task. The icon is displayed to the right of its associated task. Figure 10-7 provides an example of task and event labels and event icons. If different types of events (e.g., ones undertaken at different locations) are presented on a schedule, they are differentiated by color or shading or include an alphanumeric designator displayed on or above the icon for the event. If a coding scheme is applied to the schedule, users can access a legend or key that describes the coding technique used. No more than nine uniquely coded event icons are presented on a schedule at one time.

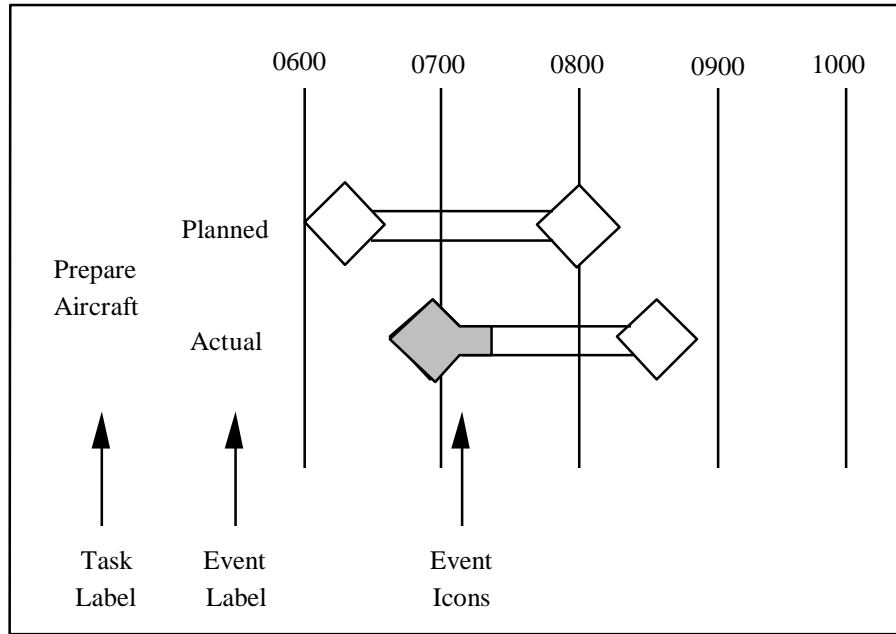


Figure 10-7. Example task and event labels and event icons.

If more than one event icon is used per task, each icon is labeled. For example, a schedule can include event icons representing planned and actual times, or earliest, latest, and actual times. Event icon labels are placed along the vertical axis or on or above the timeline. If appropriate, different scheduling attributes can be represented by displaying symbols with event icons. These symbols can be formed from various geometric shapes (e.g., circles, diamonds, squares), with fill patterns (e.g., filled symbols for events that are underway and hollow symbols for future events) used to indicate various schedule situations.

If the schedule is cluttered or users require a high degree of precision, gridlines can be used to improve the ease and accuracy with which to read information on the schedule. A gridline is displayed to indicate the present date and time; users can display or hide this line as required.

10.5.2 Schedule Manipulation

Users can define the start and stop time of the schedule displayed in the window and can do so to the desired degree of precision (e.g., day, hour). Schedule duration can be a superset of what can be displayed in the window at one time. Users can also display all or a portion of the preselected duration time. For example, users can choose to display only selected days from a schedule with a one-week duration. Users can select an individual event icon and obtain additional information about that event. The pointing device selection methods available to users conform to those listed in table 3-1.

If frequent schedule changes are anticipated, users can reschedule an event icon by directly manipulating the event icon using the object transfer methods described in section 3.5. If exact positioning of an event icon with the pointing device is difficult, the application provides alternative methods for users to locate the icon. For example, the application can allow users to enter the start and stop times for the event using text fields presented in a dialog window or provide a user definable grid that automatically repositions event icons after users place them in a new location.

11.0 USER SUPPORT RESOURCES

On-line support resources are available to users for objects and windows within an application, for the application as a whole, and for the overall system.

11.1 OBJECT-LEVEL HELP

Object-level help is displayed in the message bar of a window (if one is present) and can also be accessed using context-sensitive help.

11.1.1 Message Bar Information

If a window has a message bar, object-level help (e.g., object description, format and range of acceptable values, mandatory/optional) is displayed in that area when the object has keyboard focus. See section 8.1.2.9 for additional information on message bars.

11.1.2 Context-Sensitive Help

If context-sensitive help is available, <Shift><Help> in Motif or <Shift><F1> in Windows invokes this mode and changes the pointer to a “help” shape. Users place the pointer on a window component and click BSelect to display information about the component in a pop-up Help window. This window contains a brief description of the component and how to use it.

Motif Only: In windows with a menu bar, access to context-sensitive help is also available by selecting the On Item option in the Help menu, placing the pointer on a window component, and clicking BSelect to invoke help.

Help information is provided whenever users invoke context-sensitive help; users do not receive a "Help not available" message.

11.2 WINDOW-LEVEL HELP

Window-level help is available for every window in the application. Users access this help by activating a Help push button or menu option or by pressing <F1> (or <Help> in Motif) in an application window. The Help button or option is dimmed and unavailable for selection if help is not available; an empty help window (or one indicating that no help is available) is not displayed.

Motif Only: The Help system in CDE defines two types of help windows: a quick help window which includes a topic display area for presenting help information and an action area with one or more push buttons, and a general help window which contains a topic tree listing help topics that users can browse to find help on a subject and a topic display area. The design of quick help windows is addressed here; the design of general help windows is deferred to a future version of this document.

11.2.1 Window Design

The title of a help window, such as the one shown in figure 11-1, includes the name of the application window for which help was requested. The window is large enough to display at least ten lines of text in the display area and wide enough to display an entire line of text; the window includes scroll bars if the text exceeds the available display area. The window has an OK push button which is the default in the window and may include Previous and Next push buttons if the information can be viewed in page increments and a More push button if users can access other help resources from the window.

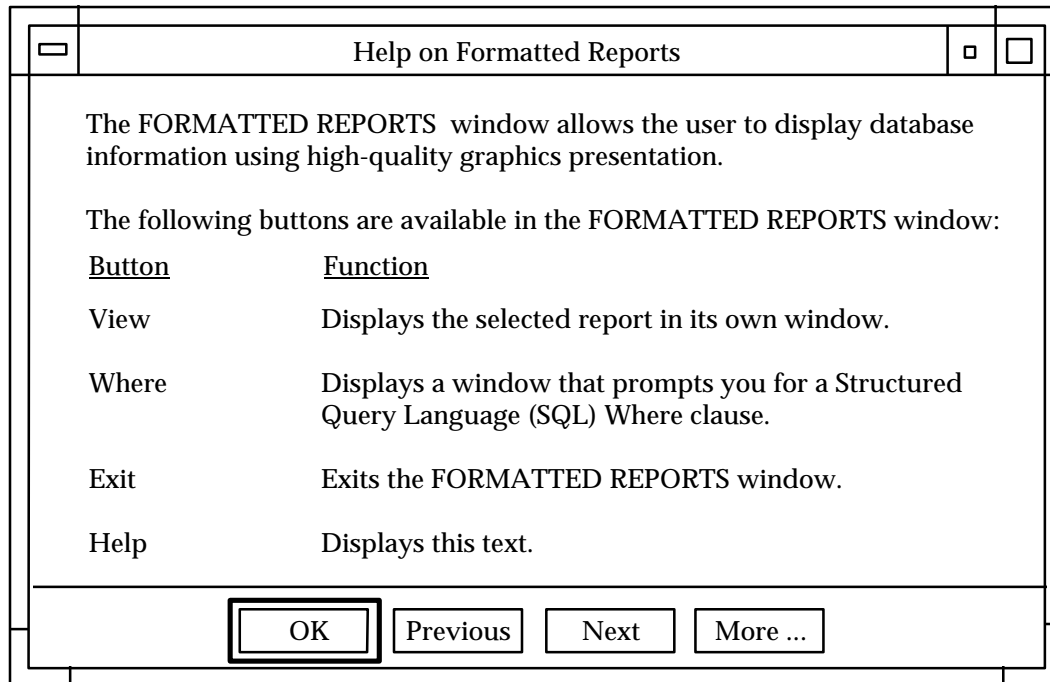


Figure 11-1. Example help window in Motif.

When a help window is displayed, it is positioned on the screen so that it does not obscure the object(s) it is describing. A help window is placed (in descending order of preference) to the right, left, above, or below the content for which help was requested. When the window is displayed, the information at the beginning of the help description is visible in the window. A help window can be moved and resized and is modeless. Users can print the content of the window, either by selecting all of the text or by marking the beginning and end of the portion of interest. A help window is removed when the parent application window is minimized or closed. When a help window is closed, focus returns to the application window for which help was requested.

11.2.2 Window Content

A help window presents only the information related to the application window for which it provides support; information on how to perform top-level tasks (i.e., require interaction with windows other than the one for which help was requested) is available in application- or system-level help support. A help window presents the following information:

- The window includes a purpose statement that describes the task(s) that users perform in the application window. Each task is described both as it relates to the functionality provided by the application and from an operational perspective. This statement can be omitted if the information is provided elsewhere in the window (e.g., in the description of actions available).
- The window explains each of the actions that can be executed in the window (excluding those pertaining to window management functions). In most cases, the actions are those available in the push buttons in the window but can also include the action options in any pull-down and pop-up menus if these are components of the window.

Depending on the size, complexity, and design of the application window to which it relates, a help window can also provide a step-by-step explanation of how users execute the task(s) presented in the window or include a More push button that provides access to this information in another help

resource. In the former case, the explanation describes both the user action, such as pressing specific keys or selecting specific controls, and the system response, such as changes in screen appearance (if any) resulting from the action.

The text in a help window is presented in mixed case, with capitalization used to emphasize or highlight significant information. Text is displayed in bulleted form, steps (e.g., in a procedure) are numbered sequentially, and explanatory information is presented in columnar form so that users can find what they are looking for quickly and easily. Graphics are used to supplement the text in a help window only if essential to understanding the task to be performed in the application window. When presenting a sequence of steps, the explanation follows the same sequence (e.g., save data before quitting; or save data, then quit the application).

11.3 APPLICATION-LEVEL HELP

Application-level help is accessed from the Help menu in the primary window(s) in the application (see section 8.1.2.3 on common menus) and may also be provided in “Read Me” files included with the application.

The remainder of this section is TBD.

11.4 SYSTEM-LEVEL HELP

System-level help is available from the CDE or Windows desktop and includes tutorials, on-line user documentation, job planning aids, and computer-based training modules. Guidelines on the conversion of existing user documentation for on-line viewing are provided in section 13.

Motif Only: Help Manager listing all of the on-line help registered on the system is available from the Front Panel. Users can access system-level support resources from this manager as well as obtain help on how to use the desktop and the Front Panel (including access to context-sensitive help on the controls, indicators, and subpanels it contains).

The remainder of this section is TBD.

12.0 ISSUES IN USER INTERFACE INTERNATIONALIZATION

Internationalization is the process of generalizing software so that it can handle multiple languages (i.e., locales) and cultural conventions without the need for re-design or re-compilation. Developers planning to field applications for use in combined or coalition warfare operations need to re-design the user interface so that it provides a “look and feel” that matches users’ expectations, interacts with users in their native language, and displays data in a manner that is consistent with users’ cultural conventions.

12.1 OPERATING WITH EXTENDED CHARACTER SETS

12.1.1 Character Rendering in Non-US Languages

Languages can be categorized in terms of the characters or symbols in which they are written. To facilitate computer processing, a character set is defined for each language to contain its written letters, numbers, and punctuation marks, with each character in the set represented by a binary value. Most European languages, including English, are based on the Roman alphabet. Because these languages contain fewer than 200 basic characters (i.e., the 26 letters in the alphabet, with upper case, lower case, and accented variations), their character sets can be encoded in a single byte. Single-byte character sets can be represented in a 16 x 16 matrix; each character is assigned a binary value ranging from 32 to 255, and the remaining values are reserved for computer control characters. ASCII (American Standard Code for Information Exchange) is the codeset in widest use in the US. All POSIX-compliant UNIX systems support this character set.

While most European languages are based on the Roman alphabet, many of them contain extended characters (i.e., ones that do not exist in English and are not available in ASCII) in their character sets. These characters include accented vowels such as é and ê; characters such as the French ç, the Spanish ñ, and the German ß and ü; and combined characters such as æ. In addition, some European languages may not use the entire Roman alphabet; Italian, for example, lacks the letter k. Despite these variations, all text in Roman-based languages is written from left to right, with each new character appended to the right of the previous character. Furthermore, the appearance of a character and its order within a character sequence do not change as new characters are entered.

While languages with fewer than 200 characters can be encoded in a single byte, complex languages such as Chinese, Japanese, and Korean each contain several thousand unique, ideographic symbols from which words are composed. Encoding such a large character set requires two bytes per character rather than one. Characters in these languages are represented by a mixture of single-byte and double-byte numeric codes ranging from 32 to 65,535. In multi-byte languages, there may be no “natural order” to characters vis-à-vis sorting, and no distinction between upper-case and lower-case forms. Because characters are composed of many strokes, text in these languages may require more space to display, and text entry may be a more complex process than in single-byte languages.

In most Roman-based languages, each character is rendered as a separate symbol of fixed shape, and characters are written in the approximate order in which they are pronounced. In other languages, however, the way in which characters are rendered graphically depends on their linguistic context. In Arabic, for example, a character may be displayed in several different forms, depending on whether it is displayed alone or as the first, middle, or last character in a word (see figure 12-1). In contextual languages, the characters that make up a symbol may be entered in several different orders, and entering a new symbol may change or even eliminate a previously entered symbol. As a result, complex algorithms may be needed to manage text line length (e.g., line breaks, justification), support the editing of individual characters within a symbol, and provide search and sort features that can recognize multiple encodings of the same symbol.

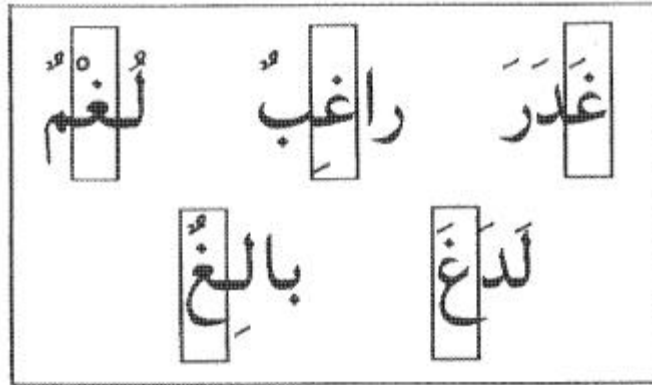


Figure 12-1. Forms of Arabic letter “G” (from Programming for the World).

Most languages are unidirectional; i.e., lines of text are presented uniformly from left to right or from top to bottom. Although Asian languages such as Chinese, Japanese, and Korean may present horizontal and vertical text on the same page, they are considered to be unidirectional because they do not mix directions in a single line of text. By contrast, Arabic and Hebrew are bidirectional; text in these languages is written from right to left, but numbers and foreign words in the same text are written from left to right. Because the direction of text entry may change from one character to the next, appropriate text handling procedures must be available in both right-to-left and left-to-right text.

12.1.2 Structural Rules for Character Handling

The application must be able to accept and process all of the characters in the character set used by the target language (i.e., the one to which the application is being converted). Because languages differ in their structural rules for character handling, assumptions made when processing a US character set may be inappropriate or inaccurate when applied to a language with extended characters. In particular, a US application is likely to require modification in order to correctly handle case conversion, ligatures, special characters, and word and character boundaries in the target language.

Case conversion. In US software, case conversion is usually performed by adding or subtracting a constant (i.e., 32) to or from the ASCII code for the character. In extended character sets, case conversion is more complicated because there is no constant difference between the numerical equivalents for upper- and lower-case representations of characters. In addition, the distinction made in English between upper-case and lower-case letters may be ambiguous or not exist at all in other languages. For example, Chinese, Japanese, and Korean have no case distinction. In other languages, an accented vowel in lower case may retain its accent in upper case, or the accent may disappear; alternatively, a vowel in upper case may or may not contain an accent in lower case, depending on the word and the language.

Ligatures. Ligatures are sequences of characters that are treated as a unit; e.g., æ is a combination of a and e. In some languages, ligatures can be entered as a single character or as two separate characters. In the latter case, both letters would be capitalized in words that are proper nouns; for example, Iceland is written as IJsland in Dutch. Ligatures occur frequently in context-dependent languages such as Arabic (as shown in figure 12-2). A US application may require revision in order to handle any ligatures that occur in the target language.

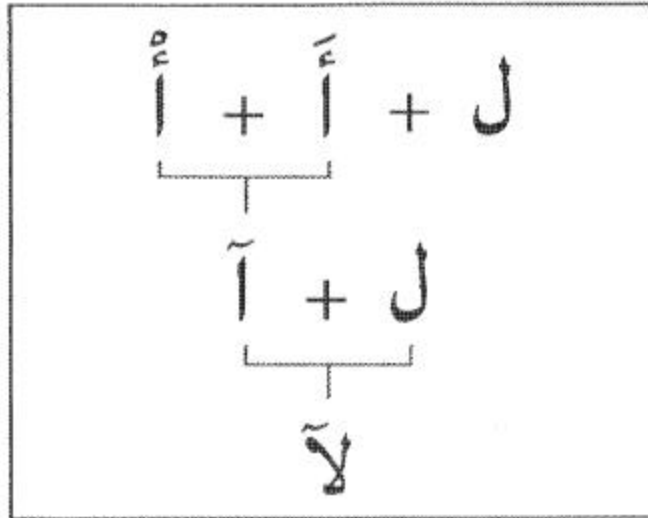


Figure 12-2. Example formation of Arabic ligature (from Programming for the World).

Special characters. Because languages differ in the meaning assigned to special characters, a US application that uses characters such as apostrophes as delimiters in a text string and restrict their use to this function may require modification when converted to certain European languages. For example, French and Italian replace the terminal vowel in an article by an apostrophe when the following noun has an initial vowel. In addition, some languages include special characters that may not be present in English or use these characters in ways that differ from US usage. For example, Spanish starts exclamations and questions with inverted exclamation mark and inverted question mark characters, while French includes a space between the last word of a sentence and a concluding exclamation or question mark. Finally, diacritical marks (i.e., the signs modifying the value or sound of characters) may have different meanings in different languages. For example, certain diacritical marks specify the doubling of consonants in Arabic but may indicate pitch in Vietnamese.

Word and character boundaries. A word in text consists of a string of characters between delimiters. In a US application, these delimiters are usually blanks or spaces but may also include the unused portion of the Roman character set. This latter approach can be problematic when converting to a Roman-based language with extended characters. These characters need to be interpreted as part of the word and not considered as delimiters. In addition, in some languages, a blank is acceptable as a numeric or phrase separator and so would not be appropriate to use as a standard word delimiter.

12.2 TEXT TRANSLATION

12.2.1 Creating Internationalized English Text

The process of text translation begins with the creation of an “internationalized English” version of the US application. All text displayed by the application is reviewed and, if necessary, modified to ensure that it is easy to understand and use. Message text (e.g., in message windows, online help) is presented in short, simple, declarative sentences whenever possible. Excessive use of subordinate and coordinating phrases is avoided, and ideas are expressed as concretely as possible. Ambiguous language, humor, jargon, and cryptic messages are likely to cause difficulty for non-US audiences and so need to be eliminated. Likewise, compound adjectives, strings of nouns, long sentences with many ideas, and negative questions can be difficult to understand and are not used. If the application needs to explain a series of concepts, they are presented in the form of a list, rather than in a text string separated by commas.

The content of each window in the application needs to be checked for US-specific language prior to translation to minimize the likelihood of misinterpretation by the target audience. The goal is to use only those terms that are employed in the same way throughout the English-speaking world. The use of acronyms and abbreviations is limited since many are not recognized internationally and may have different meaning, depending on where they are used. Similarly, when large numbers are presented, they are written as numerals; for example, the term “billion” means one thousand million in the US but one million million in some European countries. To minimize confusion, the names of months are written out when they appear as part of a date. For example, 06/10/94 can be interpreted as June 10 or October 6, depending on the user’s experience.

The application avoids presenting examples that may be uniquely American. A generic term is used, rather than what something is called in the US. For example, “stock exchange index” is an international term while “the Dow” is specifically American and “the FTSE 100” is specifically British. The messages in the application are reviewed to determine if they may be interpreted by the target audience in ways other than intended. For example, “as soon as possible” means “immediately” in the US but “when convenient” in other countries. Terms such as “left hand” can be offensive in some cultures and need to be replaced with “on the left” or “left side” instead.

12.2.2 Translating Text and Messages

All text displayed by the application (e.g., text in windows, alerts, messages, help) is translated into the target language. If appropriate, the translation is tailored to the target language in the specific country or region that will be using the application. Translated text contains proper technical terminology, especially when the terminology may differ from conversational expression in the target language. Care is taken to maintain distinctions between terminology that may be translated into the same text string in the target language. For example, Cancel and Undo are normally translated as “annulez” in French even though these commands have distinct meanings in English. Similarly, spelling and grammar can differ among varieties of a single language and so need to be adapted accordingly.

The same terminology is used in both the user interface and documentation for the application. While the goal is to provide an accurate translation of all text into the target language, it is acceptable to use US words in the text if the language does not have an adequate vocabulary with specific translations of technical words or if the target audience is accustomed to dealing with US terminology.

The accuracy of translated messages is verified since it is possible for the text of one message to be the same as another, especially if the original messages were very similar or were worded ambiguously. In addition, the meaningfulness of the translation is checked against the situation that invoked it to ensure that the information being conveyed in the original message is also conveyed in the translated version.

Translated text is reviewed to ensure that it makes grammatical sense. Some applications construct messages from two or more substrings; for example, the name of the file being deleted is inserted into the text string requesting the user to confirm deletion of the file. While this approach may work in English, the linguistic characteristics (e.g., gender, word order, special characters) of the resulting text can be awkward or inappropriate when translated into the target language. When messages are constructed by nesting or concatenating strings, the translated text that results is frequently meaningless or syntactically impossible because words or phrases were not modified to fit the grammatical rules of the target language.

Translated text uses the same character set and font as the rest of the application. Line breaks and other format changes that may have been introduced with the translation are checked for accuracy since US rules for hyphenation, punctuation, or capitalization are likely to be different from those in the target

language. If typographic variations such as italics or boldface have been added as part of the translation, they are checked to ensure that they are suitable in the target language.

Language environments have evolved unique rules defining how elements such as title lines, bulleted lists, and footnotes are used to distinguish among levels of expression and to indicate how expressions are related. The appearance of translated text is adapted as needed to satisfy these rules.

12.2.3 Translating Documentation

The documentation for an internationalized application describes a representative sample of the internationalized capabilities provided by the software. For example, an explanation of how a sorting function works describes the kinds of sorts that are performed, explains that the current locale affects the output, and provides several examples that are representative of the locales supported.

Documentation text contains simplified English. Whenever possible, a single term is selected to express a concept, and the use of synonyms is minimized. However, these changes are made in such a way as to not reduce the precision of the text, create awkward phrasing (with an increase in overall text length), or produce unacceptably dull or boring text. References (e.g., to sample users) and examples that are specific to one culture are modified to be more international in focus. Any graphic symbols used in the documentation are reviewed to minimize the extent to which they are culture-specific. If necessary, a table is provided that lists the symbols and their interpretation. Finally, documentation sections such as glossaries and indexes is expanded as needed to help non-US readers find information. For example, glossaries define words that may have a different technical meaning or not exist at all in the locales supported by the application.

Because the order of items in a sorted list usually changes following translation, references to the position of items in the list are removed from documentation. Similarly, when collation sequences are described, the results are not described as sorted “alphabetically” since ideographic languages cannot be sorted alphabetically. Output is described as appearing “in sorted order as determined by the current locale.” Other changes needed when internationalizing documentation include ensuring that terms such as ASCII, text, byte, and character are used appropriately, describing any assumptions made about date and time formats, replacing references to Yes and No (e.g., when describing actions in response to a message) with words that are appropriate to the locale, and presenting the names of any individuals (e.g., sample users) in an order that is correct for the specific culture. Finally, lengthy text explanations (e.g., in online help, training materials, or other documentation) may need to be restructured or reorganized so that they follow the rules used by the target audience in organizing technical discussions or sequences of explanations.

12.3 TEXT INPUT METHODS

12.3.1 Keyboards and Keyboard Input

A language usually has a specific keyboard layout associated with it, and this layout may be different from the one available on US workstations. Conventions concerning the location of characters vary from language to language and sometimes from country to country within the same language. For example, the German keyboard reverses Z and Y from their positions on the US keyboard, and the Spanish keyboard has a different layout in Spain than in Latin America. In addition, languages may add, omit, or change the characters on a keyboard. For example, the British keyboard contains the currency symbol for pound instead of #, and the Spanish keyboard has ñ where the US keyboard has L. Non-US keyboards may mark each key with up to four different characters. Users press modifier keys (e.g., <Shift> and/or <Alt>) in combination with the key to enter the various characters marked on the key.

Because computers respond to specific physical keypresses regardless of what markings appear on the keys, a different keyboard may not be required when converting a US application into another language. A keyboard can be adapted by replacing the symbols on each key, either with adhesive labels or new key covers. The application then maps the individual keystrokes to the character set for the other language and displays the appropriate characters on the screen. If this approach is used, the function keys on the keyboard need to be mapped to the same actions as in the original software, and any messages generated when these keys are pressed are displayed as they were prior to the conversion.

Languages where diacritical marks are used extensively (e.g., French) usually provide keyboards that allow users to generate characters with these marks with a single keystroke. However, because English has very few accents, users with a US keyboard have to execute a combination of keystrokes in order to enter an extended character. “Dead” keys and a compose-based method can be implemented in the application so that users can produce this type of input.

With “dead” keys, the keystrokes consist of a “dead” (i.e., nonspacing) key, followed by the character (e.g., a vowel) to be displayed with an accent. A different dead key is assigned to each accent. When a dead key is pressed, a text input mode is invoked; the symbol on the key is not displayed, and the text cursor does not move. The mode is automatically disabled following the next keystroke; the appropriate dead key is pressed each time an accented character is being entered. If an invalid character (e.g., a consonant) is entered, the character is displayed without an accent, and feedback (e.g., a beep) is provided to indicate that the keystroke was invalid.

In a compose-based input method, when a predefined control key is pressed, a text input mode is invoked that forms the next two keystrokes into a single character. When the first character (e.g., a vowel) is typed, nothing is displayed on the screen. When the second character (e.g., the diacritic) is entered, the completed character is displayed, and the input mode is automatically disabled.

12.3.2 Approaches to Text Entry

Pre-edit methods of text entry. In most languages, users perform text entry by typing directly into a text field. However, if a keyboard cannot produce all of the symbols in a target language, a pre-edit step may be needed. Users type characters from the keyboard, usually into a pre-edit area, and then execute an action to convert the characters into other symbols appropriate to the language. These symbols are then displayed in the text field.

When a pre-edit step is required, text entry can be performed on-the-spot, over-the-spot, or off-the-spot. On-the-spot means that as users type, the characters appear directly in the text field which can contain both text in unconverted form and converted symbols. Although more difficult to implement, this approach is preferred because it is more similar to text entry as normally performed by users. In over-the-spot, a separate pre-edit area is provided for each text field; when users convert their input into final form, the symbols are displayed in the appropriate text field. Off-the-spot also provides a separate pre-edit area but uses the same area for multiple text fields; in this case, when users convert their input into final form, the symbols are displayed in the text field that has input focus.

When text entry includes a pre-edit step, the application provides feedback concerning the status of the input after users enter text in the pre-edit area and then execute an action to convert the input into final form. If insufficient information is available to perform the conversion, the application can prompt users to enter more pre-edit text, present them with a list of choices from which to select, or indicate that the conversion has failed. If an on-the-spot approach is implemented, the text field provides a visual distinction (e.g., a different text font or color) between original input and converted text so that users can easily distinguish between the two. If the pre-edit area is provided in a separate dialog window, the window is modeless so that users are not restricted to only performing text entry.

Text entry in languages with large character sets. Several options are available to support keyboard input in languages such as Chinese, Japanese, and Korean that have large character sets. Whatever method is selected must be able to accommodate context-specific variations within the language as users perform text entry. With each keystroke, converted text changes as needed in order to create a new compound character or add a mark to a previous character.

With the first option, the component elements of each character are marked on the keyboard. As users press individual keys, the elements are displayed. When a character is complete, it is displayed in place of its components. This method has been used to perform text entry in Chinese and Korean.

With the second option, users enter each character phonetically, and the phonetic form is automatically translated into the correct character. When more than one character has the same pronunciation, users are presented with an array of phonetically similar characters from which to choose. For example, users enter a root or radical character from the keyboard, then select additional strokes to complete the character from a set displayed on the screen. This method has been used to convert Roman characters to Chinese ideographs, and Hiragana and Katakana characters to Japanese Kanji.

With the third option, users type the decimal or hexadecimal encoded value for a character or select a value from a list. If the value matches an entry in the code set, the corresponding character is displayed on the screen.

Text entry in mixed character sets. Users may need to perform text entry in more than one character set (e.g., English and Korean) or in multiple locale-specific character sets (e.g., Kanji and Katakana). This flexibility can be provided by defining text input modes in each character set, along with a special keyboard character that allows users to toggle back and forth between the character sets as desired. Users with a keyboard where two character sets are marked on the keys select one of the modes to begin text entry. All of the typed text is interpreted in this character set. When the special character is encountered, the text mode toggles to the other character set and all subsequent input is interpreted in this set.

Text entry in bidirectional languages. Because bidirectional languages write text in both right-to-left and left-to-right directions, text entry may be performed in either direction, depending on the contents of a text field, and may require input in both directions within a single field. The application can support bidirectional text entry through the use of a text input mode that switches the direction of text entry when invoked. Users typing right-to-left text in a right-to-left text field press a special keyboard character to invoke the mode and type left-to-right text (e.g., a number) in the field. During left-to-right text entry, the text cursor remains at the boundary between the two text segments, and the left-to-right text is displayed to the left of the cursor as it is entered. Users press the key controlling text input mode a second time to shift back to right-to-left text entry. When the mode is disabled, the text cursor jumps to the left end of the left-to-right text that was just entered, and text entry proceeds again in a right-to-left direction.

12.3.3 Other Text Entry Actions

The text cursor remains visible during text entry to indicate the locus of typed input. In addition, the text cursor does not disappear from view as it moves from one character to the next in a string of single-byte and multi-byte characters. If the target language is bidirectional, the application can support multiple text cursors within a single text area, one indicating when text can be added in the current input direction and the other marking the last place where the direction of input changed. In contextual languages (i.e., where the appearance of existing text can change as new characters are entered), the text insertion point can move backward or forward as users perform text entry; in this case, the text cursor is displayed in a manner that is consistent with the movement of the text insertion point.

The arrow keys move the text cursor in the direction of the arrow regardless of the direction in which text is currently being entered. <Delete> deletes text in the direction opposite to the direction in which text is being entered.

If the application is being converted to a contextual language, it needs to define how certain keystrokes affect a compound symbol that is composed of several separate characters. For example, the application needs to determine when <Delete> cancels the previous keystroke (i.e., removes a character) or deletes the entire symbol. In addition, the application may need to limit the ability to insert or delete individual characters in a word if these actions would change neighboring characters or alter the appearance of the word in unintended or confusing ways.

12.4 INTERNATIONALIZING USER INTERFACE FEATURES

12.4.1 Text Expansion

When English text is translated into another language, the result is often longer than the original English. For example, the phrase “message popup” translates to “Nachrichtenüberlagerungsfenster” in German and “janela de sobreposição de mensagem” in Portuguese. The increase in text length may be as much as 200 percent, depending on the length of the original text. Some of this increase may result from the addition of spaces that were not present in the original text. Table 12-1 lists recommended allowances for expansion based on text length in English. This table refers to the number of characters in a message, with characters in multi-byte languages (e.g., Japanese) taking two bytes per character.

Table 12-1. Allowances for text expansion.

<u>Length of English Text</u>	<u>Additional Space Required</u>
Up to 10 characters	101 - 200 percent
11 - 20 characters	81-100 percent
21 - 30 characters	61 - 80 percent
31 - 50 characters	31 - 40 percent
51 - 70 characters	31 - 40 percent
Over 70 characters	30 percent

Note: This table was taken from the Microsoft Windows Software Development Kit -- Additional Windows Development Notes, as published in Software Internationalization and Localization: An Introduction.

Translated text may require adjustments in the horizontal spacing between specific pairs of characters. For example, in English, the characters f and i look better when displayed closer together than other pairs of characters. The horizontal spacing algorithms used by the application need to accommodate adjustments in the spacing of non-US characters, including pairs of characters (e.g., æ) that may be part of an extended character set.

The height of a line of translated text may be twice the height of the original text in English. Roman-based languages may supplement the character set with diacritical marks that extend above or below the basic symbol. In non-Roman languages, marks may be stacked two or three high, and small versions of characters may be placed above, below, or beside the primary symbols, causing wide variations in the height of each text line. Because of their complexity, ideographs require more space to display the strokes within them. For example, some complex Chinese characters may need to be displayed at least

50 percent larger than alphabetic characters in order to be readable. The minimum size for ideographs is usually 16 x 16 pixels. Translated text may also require adjustments to the vertical spacing between lines to ensure legibility and readability both when displayed on the screen and when printed. Extended characters, and in particular those with diacritical marks, may require additional spacing, especially when printed in upper case.

It is likely that the size and placement of controls in application windows will require adjustment in order to accommodate text expansion following translation. Menus and dialog windows will also need to increase in size in order to accommodate the longer text. Similarly, more vertical space may be needed in window components such as the title bar to accommodate larger character size, especially in languages such as Chinese, Japanese, and Korean. Finally, the size of the text included in the label of a window icon may need to increase to accommodate an extended character set, and the icon graphic may also contain embedded text that needs to be translated.

Each application window needs to be checked to ensure that all of the translated text fits properly within the window and that individual controls are positioned correctly within each window area. For example, when column headings are translated, the text may be longer than the data included in the column so that the heading has to be broken into more than one line of text. Similarly, when text labels are translated, the placement of the associated text fields may be altered and require repositioning in order to be properly aligned within the window.

12.4.2 Nonlinguistic Text Features

Capitalization, punctuation, and word order. When text is displayed in application windows, it follows the rules for capitalization, punctuation, and word order used in the target language. For example, in German, all nouns are capitalized, regardless of their position within a phrase or sentence. Depending on the language, quotation marks may be displayed as “quotation”, «quotation», »quotation«, or „quotation.“ Interrogatory sentences in Spanish begin with an inverted question mark and end with a question mark in normal orientation. Adjectives precede nouns in English word order but may follow nouns in other languages.

Hyphenation. Hyphenation is performed by the application in a manner that is consistent with the rules of the target language. These rules may call for changing the characters in a word when it is hyphenated at the end of a line, or placing hyphens between individual words when they extend beyond the end of a line. For example, in German, “drucken” and “heißen” become “druk-ken” and “heis-sen” when hyphenated. In French, a hyphen is added between a personal pronoun and “même” (e.g., “eux-même”) when these words extend beyond the end of a line.

Justification. The justification routines used by the application conform to the rules of the target language and may require some character-processing logic in order to do so. For example, in Asian languages such as Japanese where spaces are not used to delimit words, line breaks can occur anywhere within a word. However, because symbols are represented by a multi-byte character, line breaks cannot occur within a symbol nor can punctuation be the first character on a new line. Alternatively, other languages such as Arabic and Hindi do not allow breaks within words. In this case, the justification algorithm used by the application must accommodate this restriction and be able to produce justified text without excessive space between words.

Abbreviations. The abbreviations used in US software may have different meanings in other languages or not be used at all. For example, while # is commonly used as an abbreviation for number, this character is not meaningful outside the US. The symbol @ means “at” in the US but “each” in the United Kingdom. The abbreviations for ordinals are 1st, 2nd, 3rd, etc. in the US, but 1^o, 2^o, 3^o or 1^a, 2^a, 3^a in other languages, depending on the gender of the subject.

Typography. If the application displays text in a Roman-based character set, it supports the fonts (e.g., Times and Helvetica), sizes (e.g., 10-point, 12-point), and styles (e.g., plain, italic, bold) that are normally available in the target language. The application also accommodates any unique typographic conventions when displaying translated text. For example, stress in writing is indicated through the use of italics in English but by letter spacing or boldface in European languages. In Japanese, stress is indicated by underlining characters, putting a light gray background behind them, or writing the text in Katakana.

Reordering sorted information. Applications frequently present sets of related items (e.g., in lists, option menus) in alphabetical order. These items need to be reordered after translation according to a sorted order that is meaningful to the target audience. The most appropriate order may vary by application and depend on the information displayed in the items.

12.4.3 Data Formats

The application is able to recognize and correctly handle the range of formats that are used to express data in the target language. The labels for all data fields are modified to include the unit of measurement required for data entry. The application either converts the data format to one that is familiar to users or provides the capability to display data in alternate formats so that users can select the one that is most meaningful to them. For example, US users prefer to measure length in feet and yards while European users are more familiar with the metric system. If the content of data fields is not converted to a data format that is familiar to the target audience, then the label for the field is extended to include the data format as part of the label. If the application supports converting to and from both millimeters and inches, the number of digits stored is sufficient to prevent truncation errors during conversions.

The presentation of date and time is modifiable by users so that they can display this information in the appropriate time zone (e.g., India rather than Zulu) and modify it for other zones as needed. Numeric data is properly aligned according to the particular numerical separators and indicators used in the target language. In addition, if the application allows users to manipulate text, the different forms of tabulation available are modified as needed (e.g., allow the decimal tab to work with commas rather than periods) to accommodate the data formats used in the target language.

Number systems and formats. While Arabic numerals (e.g., 0, 1, 2, etc.) are widely accepted, some languages have their own numbering systems. In some cases (e.g., Chinese), the symbols are substitutes for Arabic numerals, while in others (e.g., Ethiopia), there are special characters for numbers such as 10 and 100.

When presenting numbers, a comma, period, space, and apostrophe can be used as separators for units of thousands. In some cases, an explicit separator is not required for numbers less than 10,000. Numbers can be grouped by thousands or ten thousands. The period, comma, and center dot can be used as separators for decimal numbers. Positive and negative numbers can be indicated by + and - symbols appearing either before or after the number, and negative numbers can be enclosed in parentheses.

Measurement systems and arithmetic operations. US users are familiar with the Imperial system of measurement in inches and fractions of an inch (e.g., halves, quarters, eighths) while users outside the US rely on the metric system which uses meters, liters, and grams. In addition, the US relies on the Fahrenheit scale for temperature while the rest of the world uses the Celsius scale. Similarly, traditions vary in the manner in which certain arithmetic operations are performed. For example, some countries have rules for rounding numbers that differ from those used in the US. In addition, accounting rules (e.g., to calculate compound interest) vary from locale to locale.

Currency. The comma, period, and colon can be used as separators for currency. Currency indicators include a number of symbols (e.g., \$, British pound, and the Japanese yen), alphabetic characters (e.g., FF, SFRs, kr), and combinations (e.g., CZ\$), and can be placed at the beginning, middle, or end of the currency expression. There can be one or no space between the currency symbol and the amount, and currency symbols can be up to four characters in length. Most currencies (except Japan) include two digits to indicate fractional money amounts.

Date and time. The hyphen, comma, period, space, and slash can be used as separators for the day, month, and year, or separators can be left out altogether. In numeric date formats, the month and day fields can be reversed, and in some cases, the year field can come first. Month and day names can be capitalized or in lower case and can be abbreviated using the first two or three letters or some other combination of letters.

The manner in which a date is expressed can be affected by the calendar system being used. While dates are usually based on the Gregorian calendar, some cultures use lunar calendars or the Jewish or Arabic calendar or can express the date based on the year of accession of the Emperor, as in Japan. These calendars can include day names for more than seven days, and month names for more than twelve months. Moslem countries such as Saudi Arabia and Egypt use a calendar with 12 months but only 354 or 355 days. The first day of the week is Sunday in the US but Monday in European countries, a difference that affects the manner in which calendars are displayed.

The colon, period, and space can be used as separators for hours, minutes, and seconds. The letter h can separate hours and minutes. Both 12-hour and 24-hour notation can be used. For 12-hour notation, a.m. or p.m. can appear after the time.

Although the world is divided into 24 standard time zones, countries have the freedom to set their own times. For example, in South America, Surinam's time is 30 minutes different from that of the next zone, and Guyana's is 45 minutes different. The same time zone can have multiple names, and different time zones can share the same abbreviation. Finally, countries differ in their rules concerning daylight savings time or may not use it at all, and the hemispheres differ in when it starts and ends because the seasons are reversed.

Addresses and telephone numbers. Addresses vary from two to six lines long and can include any character used in the character set for a language. The house number precedes the street name in the US and United Kingdom but follows the street name in most other European countries. Postal codes appear in various positions and can include alphabetic characters (e.g., an abbreviation for the country), separators (usually spaces), and numbers (up to seven characters and numbers in length). In many countries, each part of an address is written on a separate line; however, in South Korea, the entire address is placed on a single line, with the specific format used varying for central cities and local areas.

Telephone numbers can contain blanks, commas, hyphens, periods, and square brackets as separators. Telephone numbers can be displayed in local, national, and international formats. Local formats vary widely. National formats can have an area code in parentheses, while international formats can drop the parentheses but add a plus sign at the beginning of the number to indicate the country code.

12.4.4 Graphics

Icons and symbols. The graphic features (e.g., icons) of an application designed for a US audience may appear strange when viewed by users outside the US. For example, a mail application that changes a mailbox graphic to indicate receipt of new mail may be unrecognizable in another culture where mailboxes have a different appearance or may not be used. Certain images, colors, and numbers of objects in a group may evoke a negative reaction in another culture so that they obscure or contradict

the message they are intended to convey. As a result, the icons used in the application may need to be revised in order to match the image or symbol to the culture in which the application will be used.

Application icons use existing international symbols whenever possible. When a new symbol is created, it represents a basic, concrete concept because concrete icons require less explanation than abstract ones. In addition, each new symbol needs to be compared with existing symbols to ensure there are no conflicts. The use of stars and crosses as part of the symbol is avoided. Text is not included in an icon because it will need to be translated and may not fit into the icon when presented in the target language.

Drawings. Application windows presenting information in graphic form (e.g., line graphs, bar charts, histograms, flowcharts) incorporate translated text, including appropriate adjustments in data formats as needed. The size of the graphics objects may need to be enlarged to accommodate the increased length of translated text. Alternatively, application graphics can be modified to place text adjacent to, rather than within, the object so that changes in text length do not affect size of individual objects or the overall illustration.

Graphic design conventions vary from culture to culture. For example, Japanese artists tend to draw tables of data differently than Western artists do. As a result, the application may require modification to accommodate these conventions.

Tactical graphics. When tactical graphics are presented (e.g., in a map window), users can access a variety of map features in order to customize the display to match their preferred mode for viewing and interpreting this information. For example, where US users are likely to display road features for navigation in urban areas, Korean operators may prefer to see neighborhood names as key map landmarks.

Visual cues for alerting. There are cultural differences associated with the meaning of certain graphics used for alerting (e.g., gestures such as waving one's hand) that may be offensive to members of the target audience. The specific visual signals used by the application, especially for alerting, need to be reviewed to ensure that they convey the desired meaning in the target culture and that their representations within the software will not be objectionable to users. In addition, alert and warning messages can be supplemented with icons so that the application communicates critical information in both text and graphic form.

12.4.5 Keyboard Interaction

Mnemonics and accelerators. When menu options are translated, any mnemonics or keyboard accelerators included with the options also need to be modified so that they contain letters that reflect the translated text. In general, the guidelines for mnemonics in languages with single-byte character sets also apply to languages with multi-byte character sets, except for how mnemonics are displayed. Applications translated from the former to the latter can retain the mnemonics used in the single-byte version, with the mnemonic displayed in parentheses following the text of the menu option. If all of the characters in a menu option have been assigned as mnemonics or if the choice consists of multi-byte characters, the application can use another letter or keyboard character. The same mnemonic is assigned to an option whenever it appears in an application menu.

The layout of the user's keyboard needs to be considered when selecting the key combinations for the mnemonics and accelerators to assign to translated menu options. First, keys are selected to minimize the disruption or relearning required to execute a mnemonic or accelerator, especially a frequently used one. Second, there are no conflicts between the key combinations for entering accented characters (e.g., if a US keyboard is being used) and those being used for mnemonics and accelerators. Finally, some non-US keyboards contain only one <Alt>, located either on the left or right side of the

keyboard. The ease with which users can execute the key combination for a mnemonic is considered if one of these keyboards is being used.

Speed search and text search. If the application is translated into a language that contains accents on the first letter of words, users are able to execute a speed search (e.g., in list boxes) by typing an unaccented upper-case or lower-case letter, and the search finds instances of both unaccented and accented first letters.

If the application uses wild card characters (i.e., @, #, ?, and *) to perform text searches, it needs to determine if these characters are assigned special meaning in the target language. If necessary, an alternate set of wild card characters is selected to eliminate any possible confusability when the application is converted to the target language.

12.4.6 Text Manipulation

Sorting and collation. Applications make use of linguistic sort sequences to order the contents of alphanumeric lists or to add new information to an already sorted list. In US software, characters are usually compared according to their binary value in the code set, with characters ordered based on these values. However, variations are frequently required to reflect linguistic conventions since the binary sequence of characters may not match the linguistic sequence for the language. For example, variations may be needed to handle characters with functional equivalence (e.g., Mac and Mc usually appear together) and to address situations where a character should be ignored (e.g., re-locate and relocate should be placed together). In addition, where US software typically provides a single sorting algorithm to accommodate such variations, other languages usually support multiple sort orders. As a result, the application needs to provide users with the ability to choose a sort order that meets their needs.

Sorting rules for European languages must be able to handle extended character sets and language-specific conventions, independent of the binary values assigned to characters. These languages may contain letters after “z” or sort letters out of the standard alphabetic sequence used in the US. For example, some of these languages contain double characters that sort as one combined character, or a single character that is treated as a double character. In Spanish, double characters such as “ch” and “ll” sort as a single character, and in German, ß is a single character that is treated as “ss” when found in a word. Madell, Parsons, and Abegg provide the following examples of differences in sorting order based on ASCII and German rules, and ASCII and Spanish rules:

<u>Sorted by ASCII rules</u>	<u>Sorted by German rules</u>	<u>Sorted by ASCII rules</u>	<u>Sorted by Spanish rules</u>
Airplane	Airplane	chaleco	cuna
Zebra	ähnlich	cuna	chaleco
bird	bird	día	día
car	car	llave	loro
ähnlich	Zebra	loro	llave
		maíz	maíz

In the case of complex (i.e., multi-byte) languages, expressions can be written in a mixture of character sets. For example, the Japanese word for “water” may be written as a single Kanji character, as two Hiragana characters, as two Katakana characters, or as the four-letter Romaji expression “mizu.” As a result, sorting algorithms in these languages must be able to accept multiple character patterns as representing the same expression. These algorithms can combine a sorting order among the character sets with a sorting order for expressions within each set. In addition, the application may need to provide a sort order based on a symbol feature that is not captured within the character code. For example, Chinese expressions may need to be sorted by the numeric value of the character as

represented in the coded character set as well as by the number of strokes required to represent the character, the radical (i.e., root) of the character, or the number of strokes added to the radical. Finally, the application may need to implement a sort order based on the way symbols are pronounced. In this case, each symbol may have to be stored in both graphic and phonetic form, with the resulting sort order listing symbols that are phonetically similar but visually different near each other.

Editing functions. Editing functions (e.g., search and replace, cut and paste, and spell checking) can accommodate the unique features of the target language, including instances where the appearance of a word changes when it is hyphenated, where it appears in lower rather than upper case, or where it contains a combination character such as æ. In contextual languages such as Thai, the characters that make up a compound symbol may be entered in several different orders, with the appearance of the symbol varying based on the order in which the characters are entered. In other languages (e.g., Greek), the appearance of a character can vary depending on its position in a word. If the application performs string searches in these languages, it is able to recognize any of several possible character sequences and judge them to be the same or different as appropriate.

12.4.7 Adjustments for Bidirectional Languages

If the application is being converted to a bidirectional language such as Hebrew or Arabic, it implements a right-to-left screen orientation. Window appearance for these languages is the mirror image of the corresponding US (i.e., left-to-right-oriented) window, except that the placement of the Window menu, Maximize, and Minimize buttons in the title bar does not change. Window titles, headings, messages, and controls are translated, except for English acronyms and key names (e.g., F1, Alt) and key combinations (e.g., Shift+Del).

In general, all of the information in a window is displayed in an orientation that is correct for the user. With respect to information placement, the specifications in this style guide apply, except that “right” and “left” are interchanged. However, physical right and left remain the same. As in unidirectional languages, <Left> and <Right> move the cursor in the direction of the arrow indicated on the key; the right and left buttons on the pointing device behave as defined in this document; and left and right movement of the pointing device moves the pointer in these directions. If the application chooses to mix both right-to-left and left-to-right elements within the same window, it follows the relevant specifications defining information display for unidirectional and bidirectional languages.

12.4.8 Printing

Peripheral devices such as printers are capable of handling the character set for the target language. The full character set can be loaded on the printer, and the printer can produce all of the extended characters required by the language.

While the US standard paper size is 8.5 x 11 inches, most countries use ISO A4 size which is slightly longer and narrower than the US standard. As a result, printer capabilities (e.g., different paper trays) may need to be adjusted in order to handle the standard paper and envelop sizes used by the target audience, and the application may need to be modified to handle the varying sizes page layouts dictated by the different paper sizes. For example, hardcoded rules regarding paper margins are removed, and users are allowed to specify how they want text to appear and to do so using measurement units with which they are familiar.

Adjustments made in window format to accommodate text expansion also need to consider text presentation when the content of the window is printed. In particular, the amount of vertical space between lines of text is sufficient to print all extended characters, including those with accents, in both upper and lower case.

13.0 ON-LINE USER DOCUMENTATION

This section defines a generic structure for on-line user documentation produced in HyperText Markup Language (HTML) format and viewed with a World Wide Web browser such as Mosaic or Netscape. User documentation includes text-based documents such as user manuals as well as graphics-based documents containing collections of images such as briefing slides. This section addresses the conversion of existing documents into an on-line format that supports both browsing and printing; as a result, it may have limited applicability to the creation of new documentation designed solely for on-line access. In addition, this section defines a generic format for the presentation of information in HTML-based pages⁴⁷ but does not prescribe a specific set of markup tags to be used in delivering documents in this format.

13.1 DOCUMENT LIBRARY CONTENTS

These guidelines assume the presence of a document library homepage from which individual documents are accessed. The Library Contents page, shown in figure 13-1, lists the title of each document and a short (2-3 sentence) description of its content. Alternate formats (e.g., PostScript) of the document available in the library are listed, along with a link to each format.

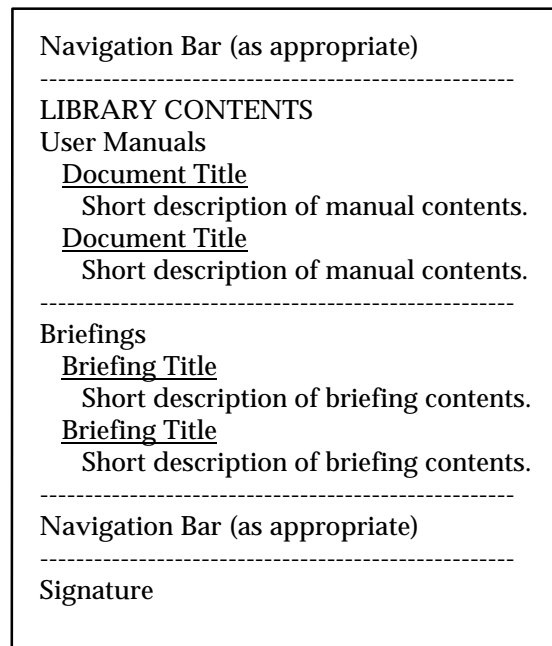


Figure 13-1. Format for a Library Contents page.

Each title links to the document itself and displays the Document Title page when selected. Any additions to or changes in the Library Contents page are flagged with small graphic icons (such as a yellow “new” burst) and not by flashing the document title or other information about the document. Groups of related documents (e.g., user manuals, briefings) are listed together and identified by a heading. Horizontal separators are used to delimit these groups and to set apart the navigation areas from the rest of the page.

⁴⁷ In this section, “page” refers to the contents of the file currently displayed in the browser window. When this term is used to describe the unit of printed output, it is identified as such.

The title of the page (i.e., the text that appears in the title bar of the browser window) is the name of the document library (e.g., JMCIS User Documentation). If appropriate, navigation bars provide links to other homepages (e.g., an on-line version of Jane's Fighting Ships) available from the browser. The final entry on the page is a signature identifying the author (e.g., name and e-mail address), creation date, and last-modified date for the page.

13.2 TEXT-BASED DOCUMENTS

13.2.1 Elements of Text-Based Documents

The guidelines presented here assume that a text-based document consists of (1) front matter such as a title page, foreword, and contents, (2) the main body of the document, and (3) end matter such as references, document-specific appendices, acronyms and abbreviations, a glossary, and an index. The information in the body of the document is organized hierarchically into chapters, sections, topics, and subtopics, with each level identified by a decimal-based numbering scheme, as shown in figure 13-2.

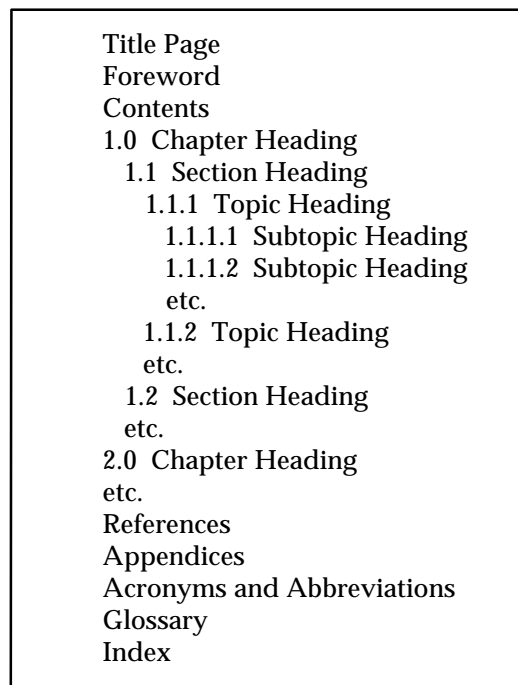


Figure 13-2. Generic structure of text-based documents.

In the guidelines that follow, the on-line structure of text-based documents depends on the overall length (in number of printed pages) and complexity (i.e., extensiveness of front and end matter) of the document. Documents can be categorized as single-part, simple multi-part, or complex multi-part. Single-part documents are those whose total length is five printed pages or less. The front and end matter in such documents is usually minimal, with contents and perhaps references included with the document body. The on-line version of single-part documents provides access to all of the information in the document from a single page in the browser window.

Simple multi-part documents are those whose total length is 100 printed pages or less, while complex multi-part documents are those that exceed 100 printed pages in length. In both cases, front and end matter are usually extensive. In simple multi-part documents, the contents may be 1-2 printed pages in length while this part of a complex multi-part document is usually much longer. The on-line

version of the former provides links to all parts of the document from a single Document Contents page, while the on-line version of the latter provides this access from separate Document Contents and Chapter Contents pages.

13.2.2 Single-Part Documents

A single-part text document is composed of Document Title and Main Body pages.

Document Title page. The Document Title page in a single-part document, shown on the left in figure 13-3, presents the full title of the document, ancillary document identification such as version number, document number, and publication date, and the name and address of the organization releasing the document. If a graphic image such as an organizational logo is included, it follows the ancillary document identification on the page. The document title serves as a link to the document itself.

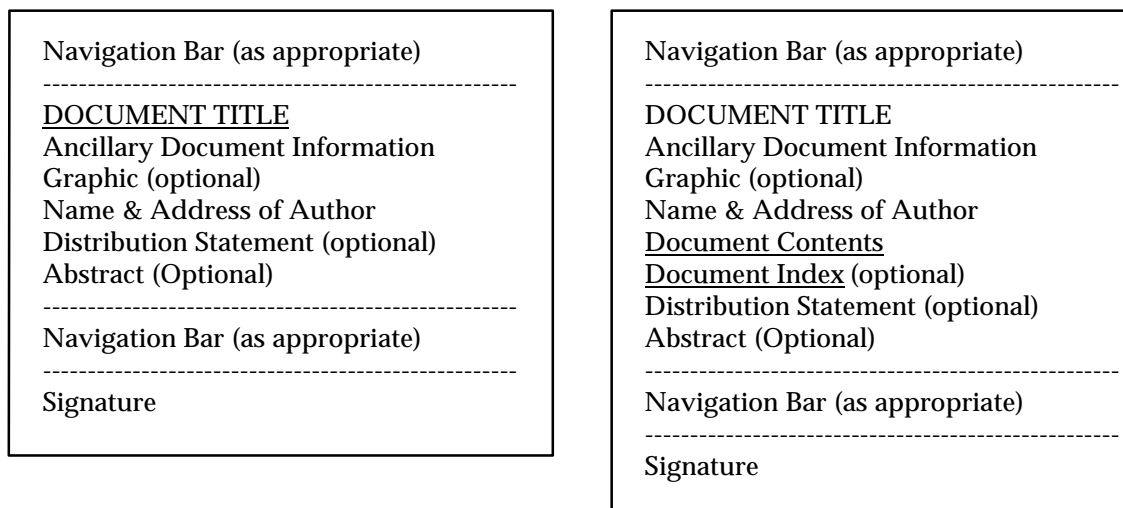


Figure 13-3. Format for a Document Title page in a single-part and multi-part document.

The information on the page (including the graphic, if one is used) is arranged so that it fits within the browser window when the window is sized to the portrait view of a printed page. If desired, a distribution statement can be included on the page, and an executive summary or abstract can be provided if one is available for the document. The navigation bars provide links to the Library Contents page and other homepages as appropriate. A signature is included at the bottom of the page, with horizontal separators delimiting it and the navigation areas from the rest of the page.

Main Body page. The Main Body page of a single-part document, shown in figure 13-4, includes the document title, followed by the document contents, front matter (as available), the document body, and end matter (as available). The contents provides links to each piece of front matter and end matter and all of the chapter, section, topic, and subtopic headings in the document. When a link is selected, the page scrolls to that portion of the document, with the heading that was selected positioned at the top of the browser window. The page includes navigation bars with links to other parts of the document library, as appropriate, and a signature at the bottom of the page. Horizontal separators are used to delimit the different parts of the document within the page.

Navigation Bar

Document Title:

CONTENTS
<u>Front Matter</u> (if available)
<u>1.0 Chapter Heading</u>
<u>1.1 Section Heading</u>
<u>1.2 Section Heading</u>
etc.
<u>End Matter</u> (if available)

Document Body (text and images)

End Matter Information (as available)

Navigation Bar

Signature

Figure 13-4. Format for a Main Body page in a single-part document.

13.2.3 Multi-Part Documents

A simple multi-part text document is composed of Document Title, Document Contents, Main Body, and Front and End Matter (if available) pages. A complex multi-part text document includes Document Title, Document Contents, Chapter Contents, Main Body, and Front and End Matter (if available) pages.

Document Title page. A Document Title page, shown on the right in figure 13-3, is provided for both simple and complex multi-part documents. The format and content of this page is the same as that for single-part documents, except that the page provides a link to a Document Contents page (rather than using the document title as the link to the document) and, if appropriate, to an Index page for the document.

Document Contents and Chapter Contents pages. The Document Contents page in a simple multi-part document, shown in figure 13-5, lists the individual pieces of front matter and end matter and all of the chapter, section, topic, and subtopic headings in the document, with each entry on the page linking to the information in that part of the document. The subordinate headings in each chapter are indented to provide a visual indication of their hierarchy within the chapter. The page is formatted to provide correct indenting of multi-line headings and to support printing in portrait view without cropping any of the text on the page. It is assumed that each heading will be numbered to indicate its level within the hierarchy of information in the document; as a result, the use of markup tags that display text as a numbered or unnumbered list is not recommended since they add either bullets or numbers to the text that follows.

Navigation Bar (as appropriate)

Document Title:

DOCUMENT CONTENTS
<u>Available Front Matter</u>
<u>1.0 Chapter Heading</u>
<u>1.1 Section Heading</u>
<u>1.1.1 Topic Heading</u>
<u>1.1.1.1 Subtopic Heading</u>
<u>1.1.1.2 Subtopic Heading</u>
<u>1.1.2 Topic Heading</u>
<u>1.2 Section Heading</u>
etc.
<u>2.0 Chapter Heading</u>
etc.
<u>Available End Matter</u>

Navigation Bar (as appropriate)

Signature

Figure 13-5. Format for a Document Contents page in a simple multi-part document.

The Document Contents page includes navigation bars with links to other parts of the document library as appropriate. Document identification (i.e., the document title) is available at the top of the page (below the navigation bar, if one is included), and a signature provided at the bottom of the page, with horizontal separators used to delimit these areas on the page.

The Document Contents page for a complex multi-part document, shown on the left in figure 13-6, differs from this page in a simple multi-part document in that it includes only chapter-level headings, with each heading linking to another page where the subordinate headings for that chapter are presented. The remainder of the Document Contents page is formatted in the same manner as this page in a simple multi-part document.

Navigation Bar (as appropriate) ----- Document Title: ----- DOCUMENT CONTENTS <u>Available Front Matter</u> <u>1.0 Chapter Heading</u> <u>2.0 Chapter Heading</u> etc. <u>Available End Matter</u> ----- Navigation Bar (as appropriate) ----- Signature	Navigation Bar ----- Document Title: Chapter Title ----- CHAPTER CONTENTS <u>1.0 Chapter Heading</u> <u>1.1 Section Heading</u> <u>1.1.1 Topic Heading</u> <u>1.1.1.1 Subtopic Heading</u> <u>1.1.1.2 Subtopic Heading</u> <u>1.1.2 Topic Heading</u> <u>1.2 Section Heading</u> etc. ----- Navigation Bar ----- Signature
---	---

Figure 13-6. Format for Document Contents and Chapter Contents pages in a complex multi-part document.

The Chapter Contents page, shown on the right in figure 13-6, provides links to the information in each section, topic, and subtopic in that chapter of the document. The page includes both document and chapter identification, with the remainder of the page formatted in the same manner as a Document Contents page. The navigation bar provides links to the Document Contents and Document Index pages for the document and to other library pages as appropriate.

Any page numbers that are a carryover from the paper version of the document are deleted from a Document or Chapter Contents page (since browsers do not provide page numbering when a page is viewed on-line or printed). The portion of the contents that lists the figures and tables in the document is also deleted (unless it is important that users of the document be able to access individual figures and tables, in which case these lists are retained and each entry in the list is defined as a link to that information within the body of the document).

Main Body page. The Main Body pages in a multi-part document, shown in figure 13-7, include document and chapter identification immediately below the navigation bar at the top of the page and signature information below the navigation bar at the bottom of the page. The first heading on the page uses the markup tag that matches the level of the information in the document (rather than default to an <h1> tag because it is the first heading on the page).

Navigation Bar

Document Title
Chapter Title

HEADING
Text and Images

Footnote Text (if any)

Navigation Bar

Signature

Figure 13-7. Format for a Main Body page in a multi-part document.

Front and End Matter pages. A Front Matter page, shown on the left in figure 13-8, includes the document title followed by the name of the front matter (e.g., (Foreword) and the text (and any images) in that part of the document. The page provides navigation bars with links to the Document Contents and Document Index pages, and a signature at the bottom of the page.

Navigation Bar	Navigation Bar
-----	-----
Document Title:	Document Title:
-----	-----
FRONT MATTER HEADING	END MATTER HEADING
Text (and images, as appropriate)	<u>A</u> <u>B</u> <u>C</u> <u>D</u> <u>E</u> <u>F</u> <u>G</u> <u>H</u> <u>I</u> <u>J</u> <u>K</u> <u>L</u> <u>M</u> <u>N</u> <u>O</u> <u>P</u> <u>Q</u> <u>R</u> <u>S</u> <u>T</u> <u>U</u> <u>V</u> <u>W</u> <u>X</u> <u>Y</u> <u>Z</u>

Navigation Bar	A
-----	Entry and definition/section number
Signature	etc.
	B
	Entry and definition/section number
	etc.
	etc.

	Navigation Bar

	Signature

Figure 13-8. Format for Front Matter and End Matter pages.

An End Matter page, shown on the right in figure 13-8, includes the document title followed by the name of the end matter (e.g., Acronyms and Abbreviations, Glossary, Index) and a set of alphabetic links that, when selected, scroll to the heading for the text beginning with that letter. The navigation bar on the Acronyms and Abbreviations and the Glossary pages provides links to the Document Contents and the Document Index pages while the navigation bar on the Index page includes a link to the

Document Contents page. Each End Matter page contains signature information below the navigation bar at the bottom of the page.

Any page numbering that is a carryover from the printed version of the document is deleted from the entries on the Index page and replaced with the hierarchical number (e.g., 1.2.3.1) for the part of the document where the entry is addressed. If the entry is related to more than one part of the document, each instance is listed in numerical order and defined as a link to the page for that part of the document.

13.2.4 Information Presentation and Navigation

13.2.4.1 Page Size in Multi-Part Documents

The following options are available for defining the amount of information available on a page in a multi-part document:

Option 1: When one of the headings in a chapter is selected, the page displayed contains all of the information in the chapter, with the page scrolled so the heading selected is positioned at the top of the browser window. One disadvantage of this option is that the time to load a long chapter may be excessive, especially if it contains a large number of images; also, if the chapter is printed, the amount of information on the page may exceed capability of the browser to print in its entirety (e.g., previous versions of Netscape were able to print a maximum of about 25 pages).

Option 2: When one of the headings in a chapter is selected, the page displayed contains only the information at that level in the hierarchy (e.g., selecting a section heading loads the information for all of the topics and subtopics within that section; selecting a topic heading loads the information for all of the subtopics within that topic). This option has the same disadvantage as option 1 if a large amount of information is included on the page.

Option 3: When one of the headings in a chapter is selected, the page displayed contains only the information between that heading and the next one for which a link is defined (e.g., selecting a section heading loads the information between that heading and the first instance of a topic heading in that section; selecting a topic heading loads the information between that heading and the first instance of a subtopic heading in that topic). The disadvantage of this option is that some headings may lack text (so the page is empty) or may be uninformative in their content (so that the page provides introductory material for information appearing on a different page).

The option that is most appropriate for the length, structure, and content of the document is selected and implemented throughout the document. In general, the guidelines presented here recommend that an HTML page contain a maximum of five printed pages of information. A simple, multi-part document may be short enough to use option 1 whereas a complex, multi-part document may need to use option 2 or 3. Regardless of the option selected, the amount of information on each page is able to stand alone both in its structure and content. It is inappropriate to assume, for example, that the page for Section 1.1 of a document will be viewed before the page for Section 1.2. Similarly, it is inappropriate to begin the text on a page with "The next step is ..."

13.2.4.2 Text, Tabular, and Graphic Information

The following markup tags are recommended for the headings in a document: Chapter headings in upper case letters and tagged <h1>, section headings in upper case and tagged <h2>, topic headings in upper case and tagged <h3>, and subtopics in mixed case and tagged <h3>. If a different mapping is used, the heading levels are used in order (e.g., an <h1> tag is followed by an <h2> tag but not by an <h3> tag).

The document author should assume 12-point Times as the default proportional font for displaying the primary type of text in the browser window, and 10-point Courier as the default fixed font for displaying the secondary type of text in the window. The default proportional font is used for any text included in the graphics for the document; the default fixed font is used for any preformatted text (e.g., to present tabular information), with line length defined so that the text fits on a page when printed in portrait view.

These guidelines do not require that links be defined from the main text on the page to other parts of the document; these links can be provided at the discretion of the document author to be available as appropriate to the structure and content of the individual document.

If the page contains classified information, each paragraph is marked with the appropriate classification level. If the printing capabilities of the browser do not support the addition of headers (so that page-level classification markings can be printed), these markings are added manually.

Footnotes are numbered consecutively within each page and displayed as normal text, not as a footnote tag (doing so makes the footnote number a link to the text at the end of the document). The text for the footnotes is presented following the main text on the page and delimited by a separator.

Tabular information is formatted to fit within the portrait view of a printed page and included within the main text on the page, rather than presented in an external file. Tables are numbered sequentially within each chapter and placed immediately following the paragraph in which the table is first referenced. When converting an existing document for on-line access, care should be taken to use markup tags that will correctly align the information in the table.

Whenever possible, graphics are sized to fit within the portrait view of a printed page and displayed as in-line images on the page. Graphics that have to be sized to fit a landscape view are opened as an external image. These graphics are not displayed as in-line images since they require resizing of the browser window in order to be viewed in their entirety and result in hardcopy that is printed in landscape, rather than portrait, view.

When graphics are opened as an external image, they are displayed as a separate page in the current window and obscure the text information to which they relate. This approach to presenting graphics is not used since it interferes with the user's ability to view all of the information (i.e., both text and graphics) related to a particular topic in a single window.

Graphics have the same background color as the rest of the browser window and are numbered sequentially within each chapter. The graphic (or an icon indicating the presence of an external image) is placed immediately following the paragraph in which it is first referenced. The caption for the graphic follows the graphic and is part of the main text on the page (rather than included with the image). Graphics that parent (i.e., link to) other graphics provide a visual cue (e.g., a box surrounding the graphic) to indicate the presence of the link, and the child graphic is displayed as an external image.

13.2.4.3 Navigation Within a Document

The navigation bar in a multi-part document provides links to the Document Contents and Document Index pages (if the latter is provided in the document) and to the “previous” and “next” information in the document. The links to the Contents and Index pages appear at the left end of the bar, with the remainder of the links arranged horizontally to the right. The bar is the width of the portrait view of the printed page, with the text for the links allowed to wrap to a second row as needed.

In a simple multi-part document, the navigation bar include the following links:

Document	Document	Previous	Next	
Contents	Index	Chapter	Chapter	

In a complex multi-part document, the links included in the navigation bar depend on the option used to define page size in the document. If option 1 is implemented, the navigation bar includes the following links:

Document	Document	Previous	Chapter	Next	
Contents	Index	Chapter	Contents	Chapter	

If option 2 or 3 is implemented, the links in the navigation bar provide access to levels of the hierarchy at and above the one currently being viewed. For example, if the page contains section-level information, the navigation bar includes the following links:

Document	Document	Previous	Previous	Chapter	Next	
Contents	Index	Section	Chapter	Contents	Chapter	
Next						
Section						

If the page contains topic-level information, the navigation bar includes the following links:

Document	Document	Previous	Previous	Previous	Chapter	
Contents	Index	Topic	Section	Chapter	Contents	
Next	Next	Next				
Chapter	Section	Topic				

13.3 GRAPHICS-BASED DOCUMENTS

A graphics-based document includes Briefing Contents and Slide pages. The Briefing Contents page, shown on the left in figure 13-9, lists the title of the briefing and ancillary information such as the briefing date and/or audience, followed by a list of slide titles, each of which links to an external slide image. If a set of related slides share the same title because they provide successive disclosure of the slide content (e.g., slide 1 presents the first bullet of information, slide 2 adds the second bullet, slide 3 the third bullet, etc.), only the final slide in the set is listed on the page. If a set of related slides share the same title but present different information (e.g., each one provides detail on a different portion of the slide), all of the slides in the set are listed on the page and numbered consecutively (e.g., “Slide Title (1 of 3),” “Slide Title (2 of 3),” “Slide Title (3 of 3)”).

<p>Navigation Bar (as appropriate)</p> <p>-----</p> <p>Briefing Title:</p> <p>Ancillary Information</p> <p><u>Slide Title</u></p> <p><u>Slide Title</u></p> <p>etc.</p> <p><u>Slide Index</u></p> <p>-----</p> <p>Navigation Bar (as appropriate)</p> <p>-----</p> <p>Signature</p>	<p>Navigation Bar</p> <p>-----</p> <p>Briefing Title:</p> <p>Slide Image</p> <p>-----</p> <p>Navigation Bar</p> <p>-----</p> <p>Signature</p>
---	---

Figure 13-9. Format for Briefing Contents and Slide pages in a graphics-based document.

If desired, a slide index can be included on the Briefing Contents page. This index links to a page containing a miniature view of all of the slides in the briefing, similar to the Slide Sorter view available in PowerPoint. The Briefing Contents page includes navigation bars with links to the Library Contents page and other homepages as appropriate, and provides signature information at the bottom of the page.

Each slide page, shown on the right in figure 13-8, includes the title of the briefing and a navigation bar with links to the Briefing Contents page, the Slide Index (if one is available), and the previous and next slides in the briefing. The image on each page is sized to fit in landscape view for printing.

14.0 BASIC FUNCTIONALITY IN COMMON SUPPORT APPLICATIONS

This section describes basic user interface functionality that should be available in a variety of common support applications. These specifications are provided to assist in “best of breed” selections and are not included in the UIS checklist.

14.1 WORD PROCESSING

The primary window in a word processing application is wide enough to display an entire line of text without having to scroll the window horizontally. The window provides scroll bars for scrolling vertically through the document, and a standard location (e.g., the message bar of the window) where users can determine their current location within the document (e.g., current page number).

Users can save, access, retrieve, and rename text documents as well as print the document. Users can specify the format of a document (e.g., margins, tabs) and select the font type, size, and style (e.g., bold, italics) for text. Automatic line break, word wrap, and automatic pagination, with page numbers based on the number entered by users, are available. The text that is displayed in the window is either formatted to reflect how the document will appear when printed, or users have the option to display the text in this format before printing the document. A copy of the original document file is retained until users confirm that it is to be changed; the file is not modified automatically as users make each editing change.

The application provides both search and search/replace capabilities for users. In the former case, users type the text string to be searched, and the application locates and highlights the first instance of the string in the text. In the latter case, users type both the text string to be searched and the text string to which it is to be changed. Users have the option for the search to be either case-sensitive or case-insensitive.

14.2 GRAPHICS

The primary window in a graphics application includes a palette of drawing tools to support efficient manipulation of objects by users. Users can select and edit the attributes of objects (e.g., color, line thickness, font size), change object sizes (enlarge or reduce), and fill the object with colors or patterns. The application automatically aligns objects to an invisible rule or grid structure, completes figures (e.g., closure of a polygon), and draws lines between user-specified points. Object selection and transfer methods (e.g., drag and drop) are available and implemented according to specifications presented in this style guide.

Users can draw objects such as lines, rectangles, ovals, and arcs; these objects are displayed as they are drawn and are easy to reposition, duplicate, and delete. Users can group separate objects into a single object (e.g., combine text and a rectangle into a complex object). The objects that are displayed in

the window are either formatted to reflect how they will appear when printed, or users are provided with an option to display the objects in this format before printing them. A copy of the original graphics is retained until users confirm that the objects are to be changed; the objects are not modified automatically as users change them.

14.3 BRIEFING SUPPORT

The primary window in a briefing support application contains a display area for viewing and editing the slides in a briefing, and a set of text, drawing, and graphing tools in the form of a palette to create and manipulate objects on a slide. The window may also include an icon bar for quick access to commonly used commands for slide manipulation (e.g., to select font and color scheme, to change view scales). The window includes the name of the briefing and displays the current slide number in a standard location (e.g., the message bar of the window). The window provides vertical and horizontal scroll bars for viewing the content of the display area when entire slide does not fit in the window.

Graphic manipulation features available in the application include the ability to edit objects (e.g., cut, copy, paste, duplicate, delete), change the attributes of an object (e.g., fill, line, shadow) and the order of objects in a stack, move an object by dragging it, and group objects together to create another object. Word processing features include access to multiple fonts and type sizes, spellchecking, and search/replace. Users can select the color schemes used in each slide, including the colors for the slide background, lines and text, shadows, title text, fills, and accents.

When creating a set of briefing slides, users can specify global features (e.g., font, color scheme, graphic template) for the briefing as a whole and modify these settings for individual slides as desired. Users can import text and graphics from other sources, incorporate clip art, animation, and video as part of the briefing, and do screen captures (e.g., from the tactical display) and annotate them with text and other graphics as overlays.

Users can create a file containing each set of slides (i.e., briefing). The file content includes the individual slides in the briefing as well as the order in which they are displayed. Users are able to reorder, add, and delete slides and then save the result; they are also able to select and copy multiple slides to re-use in the current briefing or to paste into another briefing. Users can access, retrieve, and rename the briefings they have created as well as print some or all of the slides in a briefing.

The application includes an option for presenting a slide show (i.e., displaying the slides in a briefing so that they fill the screen and all of the tools, menus, and other screen elements are hidden). Users can select all or some of the slides for presentation as well as specify the slide advance, sequence, timing, and transitions for the presentation. When the slides are displayed in a presentation, all of them have the same output size, aspect ratio, and orientation (i.e., all horizontal or all vertical). As users give the presentation, they can use the keyboard or the pointing device to move between slides, stop and restart an automatic show, temporarily blank the screen, and end the show. If the pointer remains visible during the presentation, users can move it with the pointing device and use it to point to and draw on the slides.

14.4 MESSAGE HANDLING

An application supporting the preparation of military messages provides preformatted templates that conform to standard message formats; when users enter the text of a message, format control is automatic. Users can specify the data to be transmitted in the message, incorporate existing file data (including other messages received or transmitted) if desired, and save the message during preparation and upon completion. Users are provided with a basic set of message header fields and supported whenever possible in specifying the message address. For example, limited sets of frequently used terms (e.g., classification level) are available for selection, and the appropriate addressee(s) are provided

automatically when replying to a message. Users can build and maintain distribution lists of commonly used addresses and select from these lists (without having to reenter the information) when addressing messages. The address is checked for accuracy prior to transmission, with users prompted to correct any errors before the message is sent.

The application minimizes the number of user actions required to transmit a message. Users can initiate message transmission directly (e.g., by selecting a Send push button). If a message cannot be sent immediately, it is queued automatically so that users do not have to actively monitor the transmission, and undelivered messages are saved in the event of transmission failure. Users can assign message priorities and cancel or abort a transmission that has not been completed. Feedback is available on status of message transmission, confirming that messages have been sent and indicating when transmission failures occur. Users can specify what feedback they want to receive, and an automatic log of this information is maintained.

The application informs users when high priority messages are received. For example, when users log on to the system, the application provides users with a list of new messages received since they last accessed the system. In addition, during a session, the application displays an alert message on the screen to inform users of priority messages. Message notification does not interfere with ongoing system use but provides some indication of urgency if the messages have different degrees of priority.

The application automatically queues incoming messages by time of receipt and message priority and maintains logs of this information. Users can review summary information on messages that have been queued, display individual messages, save/file the ones of interest, and discard those that are unwanted. When a message is displayed, it appears in a text window, and users are able to scroll, save, and print it as they would any other text document.

14.5 IMAGERY

The primary window in an imagery application contains an imagery display area and a set of tools for manipulating the image. These tools may be presented in the form of a palette or available as sets of controls grouped according to the nature of the operation performed. The window includes identifying information about the image currently displayed, presented in a standard location (e.g., the message bar) in the window. The window provides options (e.g., vertical and horizontal scroll bars, roam and zoom functions) for viewing the content of the display area when entire image does not fit in the window.

Users can access and retrieve an image for display from a directory of images. Users can either specify the name of the image or search the directory for images matching user-defined criteria, including wildcard searches. Users are able to print (in full-resolution or low-resolution) the image currently displayed in the window.

The application provides access to a number of basic image exploitation functions. Users can roam the image (similar to panning in a map window) in both horizontal and vertical directions. Automatic, jump, manual, and patterned roam control are available, with users specifying the rate, direction, and area of interest as appropriate. As users roam an image, they can tag specific areas of interest for later recall while the image remains in the display area. Users can zoom an image, either in predefined steps or in increments they specify. Image chipping (i.e., allowing users to select and designate regions of an image for storage) is also supported. Finally, users can create and manipulate annotations to display with an image. Annotations include text, lines, icons, geometric shapes, colors, and patterns. In each case, users are able to edit, delete, reposition, resize, save, and retrieve the annotation without altering the underlying imagery data. In addition, when users manipulate (e.g., rescale) the image, the scale and orientation of the annotations are adjusted to accommodate the changes.

The application supports imagery-unique operations, including image enhancement filtering capabilities and allowing users to control the dynamic range of image data by modification of intensity and contrast. Users can select and then examine (e.g., zoom, roam) a region of interest within an image, leaving the remainder of the image unaffected. If desired, users can apply the region to the entire image. The application provides mensuration functions (for computing lengths, areas, and volumes from dimensions or angles) and performs isotropic pixel correction (i.e., converts rectangular pixels to square pixels for display purposes). Finally, users are able to create a mirror view of the image so they can adjust the image if the negative was inverted when scanned.

The application provides advanced functions for viewing geo-referenced images, including a default image rotation where “up is up” (i.e., vertical objects are oriented toward the top of the window), an automatic north rotation (i.e., with North orientated at the top of the window), and interactive rotation in user-defined increments. In addition, annotation capabilities are extended so that when users manipulate the image, the scale and orientation of the annotations are adjusted according to the type or nature of the symbol (e.g., icons do not change size but text does).

Users can plot user-selected geographic data on the image, including frame-by-frame plotting to animate the data in either a forward or reverse time direction. With respect to image fusion operations, users can register (i.e., transform an image so that it aligns with either another image or a map projection) geo-referenced images acquired from the same or different sensors and display them together.

Users can perform concurrent geometric manipulations on separate geo-referenced windows that have overlapping geographic coverage. They are able to “slave” together multiple geo-referenced windows and manipulate (e.g., roam, zoom, rotate) all the slaved windows concurrently and relatively (i.e., with the window centers maintained at a common center lat/long position, despite differences in the amount and distance coverage between windows).

INDEX

<u>Topic</u>	<u>Section</u>	<u>Page</u>
Abbreviations	9.1.3	9-2
Accelerators	3.4.2	3-10
Accelerators, availability in menus	5.5.5	5-9
Accelerators, availability in windows	8.1.2.12	8-10
Acronyms	9.1.3	9-2
Action vocabulary	Appendix C	C-1
Activation	3.4	3-10
Application design, models for	7.2	7-6
Application icons	7.2.3.1	7-12
Application management	7.1.3	7-2
Application window design	8.0	8-1
Applications providing centralized task management	7.2.1	7-6
Applications providing distributed task management	7.2.2	7-8
Assumptions	1.3	1-2
Autorepeat (Motif Only)	3.4.4	3-11
Background	1.1	1-1
Bar charts	9.2.2	9-6
Bidirectional languages, adjustments in internationalized software	12.4.7	12-14
Briefing support applications, user interface functionality in	14.3	14-1
Cancel activation	3.4.5	3-11

Capitalization	9.1.2	9-1
Check buttons	6.3	6-3
Checklist	Appendix I	I-1
Clipboard transfer	3.5.2	3-13
Color coding of information	9.4.1	9-10
Combination controls	6.10	6-12
Combo boxes	6.10.1	6-12
Command windows (Motif Only)	8.2.5.1	8-15
Commercial software, adapting controls in	6.11.2	6-13
Common Desktop Environment, implementation of	7.1	7-1
Common Desktop Environment, desktop in	7.1.1	7-1
Common Desktop Environment, icon design in	Appendix D	D-1
Common Desktop Environment, color and font sets in	Appendix D	D-1
Compliance	1.5	1-4
Compliance, style requirements for	Appendix H	H-1
Consistency in performing operations	3.6.8	3-17
Context-sensitive help	11.1.2	11-1
Controls, arrangement of	8.1.2.4	8-4
Controls, nonstandard	6.11	6-13
Controls, standard	6.0	6-1
Critical message windows (Windows Only)	8.2.4.3	8-15
Data entry windows, design of	10.1	10-1
Date/time information, format for	9.1.4	9-2
Default activation	3.4.3	3-11
Developer notes	Appendix D	D-1
Dialog windows, design of	8.2	8-10
Document library contents	13.1	13-1
Document windows (Windows Only)	8.1.2.13	8-10
Drag transfer	3.5.1	3-11
Draggable objects in windows	8.1.2.10	8-9
Drop-down combo boxes	6.10.1	6-12
Drop-down list boxes (Windows Only)	6.6.1.1	6-8
Dynamic information, presentation of	9.5	9-13
Error detection	3.6.4	3-16
Error message windows (Motif Only)	8.2.3.1	8-12
Expert activation (Motif Only)	3.4.4	3-11
Explicit destruction	3.6.5	3-16
Extended character sets, operating with	12.1	12-1
File management	7.1.4	7-3
File Open windows (Windows Only)	8.2.6.1	8-19
File Save windows (Windows Only)	8.2.6.1	8-19
File selection windows (Motif Only)	8.2.5.4	8-17
Flash coding of information	9.4.2	9-12
Flow charts	9.2.3	9-8
Focus models	3.1.1	3-1
Font list	Appendix E	E-1
Function keys, fixed	2.2.1	2-4
Function keys, variable	2.2.2	2-4
Gauges	6.8	6-10
Grammar	9.1.2	9-1
Graphical information, display of	9.2	9-4

Graphical scheduling windows, design of	10.5	10-6
Graphics applications, user interface functionality in	14.2	14-1
Graphics in internationalized software	12.4.4	12-11
Graphics-based documents, conversion for on-line access	13.3	13-10
Hardware configuration	1.3.1	1-2
Help, application-level	11.3	11-3
Help, object-level	11.1	11-1
Help, system-level	11.4	11-3
Help, window-level	11.2	11-1
Histograms	9.2.2	9-6
Imagery applications, user interface functionality in	14.5	14-3
Immediate feedback	3.6.4	3-16
Information coding	9.4	9-10
Information message windows (Motif Only)	8.2.3.2	8-12
Information message windows (Windows Only)	8.2.4.1	8-14
Information presentation	9.0	9-1
Input devices, alternative	2.3	2-6
Input devices, standard	2.0	2-1
Input focus	3.1	3-1
Interactive control	3.6	3-15
Internationalization	12.0	12-1
Keyboard input	2.1	2-1
Keyboard interaction in internationalized software	12.4.5	12-12
Labels	6.4	6-4
Latitude/longitude, format for	9.1.4	9-2
Line graphs	9.2.1	9-4
List boxes	6.6	6-7
Lists, speed and incremental search in	6.6.2.2	6-8
List-to-list transfer windows, design of	10.3	10-3
Location cursor	3.2.2.2	3-3
Map windows, design of	10.4	10-4
Menu design guidelines	5.5	5-6
Menus	5.0	5-1
Menus, availability of options in	5.5.4	5-9
Menus, common	8.1.2.3	8-3
Menus, format of options in	5.5.1	5-6
Menus, grouping of options in	5.5.3	5-7
Menus, wording of options in	5.5.2	5-7
Message bar	8.1.2.9	8-9
Message dialogs (Motif Only)	8.2.3	8-12
Message dialogs (Windows Only)	8.2.4	8-14
Message handling applications, user interface functionality in	14.4	14-2
Minimized windows, processing in	4.1.3.4	4-5
Mnemonics	3.4.2	3-10
Mnemonics, availability in menus	5.5.5	5-6
Mnemonics, availability in windows	8.1.2.12	8-10
Modes of interaction	4.1.2.2	4-4
Motif resources	Appendix D	D-1
Multiple selection list boxes (Windows Only)	6.6.1.2	6-8

Navigation	3.2	3-1
Navigation, location cursor behavior in	3.2.2.3	3-4
Navigation, text cursor behavior in	3.2.2.4	3-5
Nonlinguistic text features in internationalized software	12.4.2	12-9
Object-action selection model	3.6.1	3-15
On-line user documentation	13.0	13-1
Operational environment	1.3.3	1-3
Option menus (Motif Only)	5.4	5-6
Parent-child relationships in windows	4.1.2.1	4-3
Pie charts	9.2.4	9-9
Pointer	2.1.1	2-1
Pointer shapes	2.1.2	2-1
Pointing device buttons	2.1.3	2-3
Pointing device input	2.1	2-1
Pop-up menus	5.2	5-3
Pop-up menus, availability in windows	8.1.2.11	8-9
Pop-up text fields, availability in windows	8.1.2.11	8-9
Previewing (Motif Only)	3.4.4	3-11
Primary transfer (Motif Only)	3.5.3	3-14
Primary windows, design of	8.1	8-1
Print windows (Motif Only)	8.2.5.5	8-18
Print windows (Windows Only)	8.2.6.2	8-20
Printing in internationalized software	12.4.8	12-14
Processing modes	3.6.7	3-17
Prompt dialog windows (Motif Only)	8.2.5.2	8-16
Pull-down menus	5.1	5-1
Punctuation	9.1.2	9-1
Purpose	1.2	1-1
Push buttons	6.1	6-1
Push buttons, arrangement of	8.1.2.6	8-6
Push buttons, default	8.1.2.7	8-7
Question message windows (Motif Only)	8.2.3.3	8-12
Quick transfer	3.5.4	3-14
Radio buttons	6.2	6-2
References	1.8	1-7
Resize border	4.1.1.4	4-3
Resource sharing among applications	7.2.2.2	7-11
Reverse video coding of information	9.4.3	9-12
Scales	6.8	6-10
Scroll bars	6.7	6-9
Scroll bars, placement in windows	8.1.2.5	8-5
Secondary windows, design of	8.1	8-1
Selection	3.3	3-6
Selection dialogs (Motif Only)	8.2.5	8-15
Selection dialogs (Windows Only)	8.2.6	8-19
Selection windows (Motif Only)	8.2.5.3	8-16
Separators	6.9	6-12
Scope	1.4	1-3
Session management	7.1.2	7-2
Shape coding of information	9.4.4	9-12

Size coding of information	9.4.4	9-12
Software configuration	1.3.1	1-2
Sound coding of information	9.4.5	9-12
Spin buttons	6.10.2	6-13
Style management	7.1.6	7-4
Support applications, user interface functionality in	14.0	14-1
Surface charts	9.2.1	9-4
System design	7.0	7-1
System users	1.3.2	1-3
Tab groups, navigation in	3.2.2.2	3-3
Tabular data windows, design of	10.2	10-1
Tabular information, presentation of	9.1.6	9-3
Tactical information, coding of	9.3.2	9-10
Tactical information, presentation of	9.3	9-9
Tactical symbology	9.3.1	9-9
Tear-off menus (Motif Only)	5.3	5-4
Text-based documents, conversion for on-line access	13.2	13-2
Text entry	2.2.3	2-5
Text entry, actions in	2.2.3.2	2-6
Text entry, error checking and correction in	6.5.2.2	6-6
Text entry, modes in	2.2.3.1	2-5
Text entry, supporting effective	6.5.2.1	6-5
Text expansion in internationalized software	12.4.1	12-8
Text fields	6.5	6-4
Text font	9.1.1	9-1
Text font coding of information	9.4.6	9-12
Text information, presentation of	9.1	9-1
Text input methods in internationalized software	12.3	12-5
Text manipulation in internationalized software	12.4.6	12-13
Text searches, wild card characters in	9.1.5	9-3
Text size	9.1.1	9-1
Text style	9.1.1	9-1
Text style coding of information	9.4.6	9-12
Text translation in internationalized software	12.2	12-3
Title bar	4.1.1.1	4-1
Toolbars	8.1.2.8	8-7
Transfer	3.5	3-11
Undo capability	3.6.6	3-16
User control of interaction	3.6.2	3-15
User support resources	11.0	11-1
User-computer interaction	3.0	3-1
Virtual keys for keyboard functions	Appendix A	A-1
Virtual keys mapped to DII keyboards	Appendix B	A-2
Warning message windows (Motif Only)	8.2.3.4	8-13
Warning message windows (Windows Only)	8.2.4.2	8-14
Window components	4.1.1	4-1
Window control buttons	4.1.1.3	4-2
Window design, considerations in	8.3	8-20
Window families, behavior in	4.1.2	4-3
Window icons	4.2	4-6
Window management	4.1	4-1

Window management, considerations in	4.1.3	4-5
Window menu	4.1.1.2	4-1
Window menu bar	8.1.2.2	8-2
Window panes	8.1.2.5	8-5
Window title	8.1.2.1	8-2
Windows, arrangement of	4.1.3.2	4-5
Windows, moving between workspaces (Motif Only)	4.1.3.5	4-6
Windows, positioning of	4.1.3.3	4-5
Windows, size of	4.1.3.1	4-5
Word processing applications, user interface functionality in	14.1	14-1
Working message windows (Motif Only)	8.2.3.5	8-13
Workspace management	7.1.5	7-4

APPENDIX A

MOTIF AND WINDOWS VIRTUAL KEYS FOR KEYBOARD FUNCTIONS

Note: See appendix C for virtual keys assigned as accelerators.

<u>CDE Motif Virtual Keys</u>	<u>Windows Virtual Keys</u>	<u>Function Performed</u>
<u>Text Entry</u>		
<Insert>	<Insert>	Toggles between replace and insert mode.
<Space> <Shift><Space>	<Space>	Inserts a space in text.
<Return> ⁴⁸	<Enter> <Ctrl><Enter>	Inserts a new line in multi-line text.
<Backspace>	<Backspace>	Deletes the character to the left of the cursor.
<Delete>	<Delete>	Deletes the character to the right of the cursor.
<Tab>		Inserts a tab or moves to the next tab stop.
<u>Input Focus</u>		
<Ctrl><Esc>	<Ctrl><Esc>	Displays a window listing the currently running applications.
<Alt><Esc>	<Alt><Esc>	Navigates to the next application.
	<Alt><Shift><Esc>	Navigates to the previous application.
<Alt><Tab>	<Alt><Tab>	Navigates to the next window family.
<Alt><Shift><Tab>	<Alt><Shift><Tab>	Navigates to the previous window family.
<Alt><F6>	<Alt><F6>	Navigates to the next window in a family.
<Alt><Shift><F6>	<Alt><Shift><F6>	Navigates to the previous window in a family.

⁴⁸ The Certification Checklist in the Motif and TED Style Guides indicates <Return> inserts a new line in text while <Enter> or <Ctrl><Return> invokes the default action; appendix A in the TED Style Guide indicates <Enter> or <Return> inserts a new line in text.

	<Ctrl><F6>	Navigates to the next document window.
	<Ctrl><Shift><F6>	Navigates to the previous document window.
<F6>	<F6>	Navigates to the next pane of the active window.
	<Shift><F6>	Navigates to the previous pane of the active window.
<u>Navigation</u>		
<Tab> <Ctrl><Tab>	<Tab>	Navigates to the next tab group in a window.
<Shift><Tab> <Ctrl><Shift><Tab>	<Shift><Tab>	Navigates to the previous tab group in a window.
<Up>	<Up>	Navigates up one unit (e.g., one line).
<Down>	<Down>	Navigates down one unit.
<Left>	<Left>	Navigates left one unit (e.g., one character).
<Right>	<Right>	Navigates right one unit.
<Ctrl><Up>	<Ctrl><Up>	Navigates up one large unit (e.g., one paragraph).
<Ctrl><Down>	<Ctrl><Down>	Navigates down one large unit.
<Ctrl><Left>	<Ctrl><Left>	Navigates left one large unit (e.g., one word).
<Ctrl><Right>	<Ctrl><Right>	Navigates right one large unit.
<Home>	<Home>	Navigates to the leftmost element (e.g., the beginning of a line).
<End>	<End>	Navigates to the rightmost element (e.g., the end of a line).
<Ctrl><Home>	<Ctrl><Home>	Navigates to the top leftmost element (e.g., the beginning of data).
<Ctrl><End>	<Ctrl><End>	Navigates to the bottom rightmost element (e.g., the end of data).
<PageUp>	<PageUp>	Navigates up one page.
<PageDown>	<PageDown>	Navigates down one page.

<Ctrl><PageUp>	<Ctrl><PageUp>	Navigates left one page.
<Ctrl><PageDown>	<Ctrl><PageDown>	Navigates right one page.
	<ScrollLock>	Toggles between normal and scroll lock mode.
<u>Selection/Activation</u>		
	<F8>	Toggles between normal and extend mode.
<Shift><F8>	<Shift><F8>	Toggles between normal and add mode.
<Shift><Space> <Shift><Select> ⁴⁹		Extends a selection to the cursor position except in text (also see Range Selection in Text).
<Backspace> <Delete>	<Backspace> <Delete>	Deletes the current selection.
<Ctrl></>		Selects all of the elements in a collection.
<Ctrl><\\>		Deselects all of the selected elements in a collection.
<Space> <Select>	<Space>	Selects/activates the element with focus.
<Enter> <Return> <Ctrl><Return>	<Enter>	Invokes the default action.
<Esc> <Cancel>	<Esc>	Stops or cancels the current interaction.
<u>Menu Navigation/Activation</u>		
<Alt><Space> <Shift><Esc>	<Alt><Space>	Navigates to (i.e., displays) the Window menu in an application window.
	<Alt><Hyphen>	Navigates to (i.e., displays) the Window menu in a document window.
<F10> <Shift><Menu> ⁵⁰	<F10>	Navigates to/from a menu bar.

⁴⁹ The Certification Checklist in the Motif and TED Style Guides indicates that <Shift><Select> and <Shift><Space> extend a selection to the cursor position (except in text). Appendix A in the TED Style Guide indicates <Shift><Space> and <Ctrl><Shift><Space> perform this function.

<Shift><F10> <Menu>		Navigates to (i.e., displays a pop-up menu.
<Select> <Space>		Displays an option menu when focus is on an option button.
<Enter> <Return> <Select> <Space>	<Enter>	Activates a menu or menu option.
<u>Access to Help</u>		
<F1> <Help>	<F1>	Displays a Help window.
<Shift><Help>	<Shift><F1>	Activates context-sensitive help mode.
<u>Range Selection in Text</u> (Motif Only)		
<Shift><Up>		Extends the selection up one line.
<Shift><Down>		Extends the selection down one line.
<Shift><Left>		Extends the selection left one character.
<Shift><Right>		Extends the selection right one character.
<Ctrl><Shift><Up>		Extends the selection up one paragraph.
<Ctrl><Shift><Down>		Extends the selection down one paragraph.
<Ctrl><Shift><Left>		Extends the selection left one word.
<Ctrl><Shift><Right>		Extends the selection right one word.
<Shift><Home>		Extends the selection to the beginning of a line.
<Shift><End>		Extends the selection to the end of a line.
<Ctrl><Shift><Home>		Extends the selection to the beginning of the data.
<Ctrl><Shift><End>		Extends the selection to the end of the data.

⁵⁰ The Certification Checklist in the Motif and TED Style Guides includes <Shift><Menu> and <Menu> for navigating to/from a menu bar and a pop-up menu, respectively. Appendix A in the TED Style Guide omits these key combinations from the list of keyboard functions.

<Shift><PageUp>	Extends the selection up one page.
<Shift><PageDown>	Extends the selection down one page.
<Ctrl><Shift><PageUp>	Extends the selection left one page.
<Ctrl><Shift><PageDown>	Extends the selection right one page.

APPENDIX B

MAPPING OF MOTIF AND WINDOWS VIRTUAL KEYS TO DII KEYBOARDS

<u>Virtual Key</u>	<u>HP Keyboard</u>	<u>Sun 4 Keyboard</u>	<u>Sun 5 Keyboard</u>	<u>PC Keyboard</u>
</>	/	/	/	/
<\>	\	\	\	\
<Alt> ⁵¹	Extendchar	Diamond key	Diamond key	Alt
<Backspace>	Backspace	Backspace	Backspace	Backspace
<Cancel>	DelEsc	--- (use Esc)	--- (use Esc)	--- (use Esc)
<Copy>	--- (use Ctrl+Insertchar)	Copy	Copy	--- (use Ctrl+C)
<Ctrl>	Ctrl	Control	Control	Ctrl
<Cut>	--- (use Shift+Delchar)	Cut	Cut	--- (use Ctrl+X)
<Delete>	Delchar	Delete	Delete	Delete
<Down>	Down	Down	Down	Down
<End>	--- (use Extendchar+ Right)	End	End	End
<Enter>	Enter	Enter	Enter	Enter
<Esc>	DelEsc	Esc	Esc	Esc
<F1> - <F10>	F1 - F10	F1 - F10	F1 - F10	F1 - F10
<Help>	Help	Help	Help	--- (use F1)
<Home>	Home	Home	Home	Home
<Insert>	InsertChar	Ins	Ins	Insert
<Left>	Left	Left	Left	Left
<Menu>	Menu	--- (use Shift+F10)	--- (use Shift+F10)	---
<PageDown>	Next	PgDn	PgDn	PageDown
<PageUp>	Prev	PgUp	PgUp	PageUp
<Paste>	--- (use Shift+Insertline)	Paste	Paste	---
<Return>	Return	Return	Return	--- (use Enter)
<Right>	Right	Right	Right	Right
<Select>	Select	--- (use Space)	--- (use Space)	---

⁵¹ In X Window, the Alt key is called Meta; however, no keyboard uses a key labeled Meta. Meta is labeled Alt on 386-based personal computer, Silicon Graphics Iris, and DG Aviiion keyboards, Extendchar on Hewlett Packard keyboards, and is the key with the diamond shape (next to Alt) on the Sun keyboard.

<Shift>	Shift	Shift	Shift	Shift
<Space>	Space	Space	Space	Space
<Tab>	Tab	Tab	Tab	Tab
<Undo>	--- (use Extendchar+ Backspace)	Undo	Undo	--- (use Ctrl+Z)
<Up>	Up	Up	Up	Up

APPENDIX C ACTION VOCABULARY

<u>Term</u> <u>Action Performed</u>	<u>Motif Mnemonic/Accelerator</u>	<u>Windows Mnemonic/Accelerator</u>
About <AppName> Displays name and version information about the application.	A ---	A ---
Add Adds a new item to a set of items.	-- ---	-- ---
Again Repeats the last operation.	-- <Ctrl><A>	-- ---
Append Adds new information to the end of an item, or adds a new item to the end of a set of items.	-- ---	-- ---
Apply Executes any changes made to controls in a window but does not close the window.	-- ---	A ---
Archive Creates a backup copy of a file (e.g., on magnetic tape or fixed disk).	-- ---	-- ---
Arrange Arranges document windows in an application-specific placement.	-- ---	-- ---
Cancel ⁵² Closes a window without executing any changes made to controls in the window.	-- ---	-- ---
Cascade Arranges document windows in an overlapping placement.	-- ---	-- ---
Change Allows the user to make changes to an existing query.	-- ---	-- ---
Check Spelling	-- ---	-- ---

⁵² Cancel is used in a window when the settings selected by users are reversible (i.e., not saved when the window is removed from the screen). Close is used in a window when the settings are saved, when a new state has been defined, or when new data have been generated as a result of selections made in the window.

Compares the words in a file or text selection against a dictionary of recognized words and identifies entries not in the dictionary (e.g., misspelled words).

Clear E ---- -- ----

Removes an object from a window without copying it to the clipboard; does not compress the remaining space.

Close C <Alt><F4> C <Alt><F4> (app windows)
<Ctrl><F4> (doc windows)

Closes a window and removes it from the screen; requests confirmation if unsaved changes have been made and allows the user to save the changes.

Close C ---- C ----

Closes the current primary window and its child windows; requests confirmation if unsaved changes will be lost.

Close -- ---- C ----

Closes a dialog window without executing the control settings in the window; used only when the window performs actions that are irreversible.

Compare -- ---- -- ----

Displays information on multiple items in a set of items.

Contents -- ---- C ----

Opens a Help window and displays a list of main topics.

Copy C <Ctrl><C> (preferred) C <Ctrl><C> (preferred)
<Ctrl><Insert> <Ctrl><Insert>

Clears the clipboard and stores a copy of the object being transferred in the clipboard. The object remains at its original location.

Copy Link K ---- -- ----

Copies a link of the object to the clipboard without removing the object from its original location.

Copy To C ---- -- ----

Allows the user to copy the selected object(s) into a folder; displays a dialog window for users to select the desired folder.

Cut T <Ctrl><X> (preferred) T <Ctrl><X> (preferred)
<Shift><Delete> <Shift><Delete>

Clears the clipboard, stores a copy of the object in the clipboard, and removes the object from its original location. If the object being transferred is graphic, the space that it occupied is left blank. If the object is text, the remaining text is compressed (i.e., shifted to the left) to fill in the space.

Defaults -- ---- -- ----

Restores all values in a window to a default state defined by the application.

Delete D <Ctrl><D> D ----

Removes an object from a window without copying it to the clipboard; compresses the remaining space.

Delete -- ---- -- ----

Removes a file from a storage device; requests confirmation prior to deletion.

Delete -- ---- -- ----

Describe	--	----	--	----
Displays a detailed explanation or description of an item.				

	--	----	--	----
Duplicate				
Creates a copy of a file and prompts the user to name the file.			Creates a copy of an item and adds it to a set of items.	

Execute	--	----	--	----
Performs a function, procedure, or process associated with the window.				

File	F	--	F	----
When used as a menu title, provides access to menu options for users to work with the data in the window as a whole.				

First	--	----	--	----
Displays the first page of information.				

Help	H	<F1>	H	<F1>
		<Help>		
Displays general information about a window.				

Index	I	----	--	----
Provides an index listing topics for all help information available in the application.				

Keyboard K ---- -- ----

⁵³ The actions performed by GoBack, Restart, Review, and Suspend are taken from the DoD style guide.

Provides information on the function keys, mnemonics, and accelerators in the application.

Displays the last page of information.

Displays a dialog window for changing link properties and accessing linked objects.

Ends processing by a system; closes all windows on the workspace and stops all processing; requests confirmation if unsaved changes have been made and allows the user to save the changes.

Moves a window to the bottom of the window hierarchy.

Annotates (e.g., with an asterisk) that an item has been selected.

Enlarges a window to its maximum size.

Combines the contents of two items into a single item.

Changes a window into a window icon.

Panel

More -- ---- -- ----

Displays a dialog window listing all open document windows.

Provides information on using the pointing device in the application.

Provides information on the function keys, mnemonics, and accelerators and on using the pointing device in the application.

Moves the location of the window in the direction indicated by the pointer.

Allows the user to move the selected object(s) into a folder; displays a dialog window for users to select the desired folder.

Creates a new file or document. Clears existing data from the client area if the current window will be used; requests confirmation if unsaved changes will be lost.

Creates a duplicate window that opens another view on the active document.

Next	--	----	N	----
Displays the next page of information or the contents of the next item in a set of items.				
Next	--	----	--	----
Allows the user to switch to the next document window.				
No	--	----	N	----
Indicates a negative response to a question and removes the window containing the question.				
Occupy All Workspaces	A	----	--	----
Enables the user to place a window in all workspaces.				
Occupy Workspace	O	----	--	----
Allows a user to specify the workspace(s) in which a window appears.				
OK	--	----	--	----
Executes any changes made to controls in a window and closes the window.				
On Item	O	---	--	---
Invokes context-sensitive help mode in the window.				
Open	O	<Ctrl><O>	O	<Ctrl><O>
Opens an existing file or document; prompts the user for the file name if more than one can be opened. Requests confirmation if unsaved changes will be lost.				
Open As	--	----	--	----
Opens an existing file; prompts the user for changes to how the file is to be presented (e.g., different format).				
Overview	V	----	--	----
Provides general information about the window from which help was requested or about the application overall.				
Paste	P	<Ctrl><V> (preferred) <Shift><Insert>	P	<Ctrl><V> (preferred) <Shift><Insert>
Copies the object in the clipboard to the new location. If the object is graphic, the paste operation copies it to the location with keyboard focus. If the object is text, the paste operation copies the object to the current position of the text cursor.				
Paste Link	L	----	L	----
Pastes a link of the contents of the clipboard to the new location.				
Paste Special	--	----	S	----
Allows the user to specify the format of the data being pasted.				
Pause	--	----	P	----
Interrupts a process without changing data entries or control logic for the process; used in combination with Resume.				
Previous	--	----	--	----
Displays the previous page of information, or returns to the file prior to the current one (e.g., Previous Chart displays the chart viewed prior to the current one).				

Primary Copy	--	<Alt><Ctrl><C> (preferred) <Alt><Ctrl><Insert>	--	----
Copies the selected object(s) to the destination.				
Primary Link	--	----	--	----
Places a link from the selected object(s) to the destination.				
Primary Move	--	<Alt><Ctrl><X> (preferred) <Alt><Shift><Delete>	--	----
Moves the selected object(s) to the destination.				
Print	P	<Ctrl><P>	P	<Ctrl><P>
Initiates a process for printing the contents of a file.				
Print One	--	----	--	---
Prints one copy of a file using the default print settings previously defined by the user.				
Print Setup	--	----	R	----
Allows the user to specify the printer and settings for a print job.				
Promote	M	----	--	----
Promotes to the primary selection the current selection of an object in a collection.				
Properties	P	<Ctrl><I>	R	----
When included in a pop-up menu, displays a Properties dialog window that can be used to set the properties of the object. If the menu is popped up over a selected or unselected object, it can display the properties of the object. If the menu is popped up over the background, it can display the properties of the collection.				
Put in Workspace	T	----	--	----
Allows the user to put a link for the selected object in the current workspace.				
Redo	R	<Shift><Alt><Backspace>	R	----
Reverts an Undo; i.e., returns an object to its state after the last operation was performed.				
Reference	R	----	--	----
Provides access to summary information about application components.				
Refresh	E	<Ctrl><R> (preferred) <F5>	--	----
Redraws the contents of a window; updates the contents of the window to reflect the current state of the underlying data.				
Rename	--	----	--	----
Renames a file; prompts the user to name the file; does not affect the contents of the file.				
Repeat <action>	--	----	R	----
Repeats the most recently performed action.				
Replace	--	----	--	----
Replaces a word or character string with a different word or character string entered by the user.				
Reselect	--	<Alt><Insert>	--	----
Reselects (and highlights) all of the items in the most recently performed selection.				

Reset ⁵⁴	--	----	--	----
Cancels any changes made to the control settings in a window that have not been applied by the application, and resets the window to the state at the last time a change was applied. If no changes have been applied, the controls are reset to the state when the window was first displayed.				
Reset to Factory	--	----	--	----
Cancels any changes that have not yet been applied and resets the window to the default state as specified by the application.				
Restart	--	----	--	----
Cancels entries made in a transaction sequence and returns to the beginning of a sequence.				
Restart Workspace Manager	--	----	--	----
Stops then restarts the Workspace Manager (e.g., after the user has customized configuration files).				
Restore	R	<Alt><F5>	R	----
Displays a minimized or maximized window in its previous size and location.				
Resume ⁵⁵	--	----	R	----
Resumes a process that was previously paused (e.g., turns on a process that has been turned off); used in combination with Pause.				
Retry	--	----	R	----
Causes a process to be attempted again.				
Revert	--	----	--	----
Replaces the current file with the version that was most recently saved.				
Review	--	----	--	----
Returns to the first display in a transaction sequence so that users can make changes if desired.				
Save	S	<Ctrl><S>	S	<Ctrl><S>
Saves the currently opened file without removing the existing contents of the client area; prompts the user for a name if the file does not have one.				
Save As	A	----	A	----
Saves the currently opened file under a new name; prompts the user for the new name. Does not remove the existing contents from the client area. Prompts the user for the new name and for overwrite verification if a file already exists with the same name as entered by the user.				
Save As Defaults	--	----	--	----
Saves the current settings as the default that will appear when the window is displayed again. The settings are not applied to any selected object, and the dialog window is not dismissed.				
Search for Help On	--	----	S	----

⁵⁴ The DoD style guide recommends the use of Cancel to erase changes just made by the user and restore the display to its previous state.

⁵⁵ The DoD style guide recommends the use of Continue to resume a process without changing data entries or control logic for the process.

Opens a Help window and allows the user to search for Help topics containing specific keywords.

Select All	S	<Ctrl></>	A	----
Selects (and highlights) all objects or items in a collection.				

Feature	Advantages	Disadvantages
Select On	--	----
Allows the user to select a subset of items based on criteria specified by the user.		

Select Pasted	A	----	--	----
Selects (and highlights) the last object(s) pasted at a location.				

Send	--	<F3>	--	----
Transmits an item via a communication channel selected by the user.				

Shuffle Down	--	----	--	----
Moves the top window in a stack backward in the workspace.				

Shuffle Up -- ---- -- ----
Brings the bottom window in a stack forward in the workspace.

Size	S	<Alt><F8>	S	----
Changes the size of the window in the direction indicated by the pointer.				

Sort	S	----	--	----
Arranges a set of items in an order specified by the user.				

Start	--	----	--	----
Begins or turns on a process.				

Stop	--	----	S	----
Ends or turns off a process at the next possible point.				

Suspend	--	----	--	----
Preserves the current transaction status when users log out of the system and permits resumption of work when users later log in to the system.				

Switch To	--	----	W	<Ctrl><Esc>
Displays a list of running applications so the user can switch to another application.				

Table of Contents	C	----	--	----
Provides a Table of Contents listing topics for all help information available in the application.				

Tasks	T	----	--	----
Provides access to help information on how to use the application.				

Tile	--	----	--	----
Arranges document windows in a tiled placement.				

Tutorial	L	T
Provides access to a tutorial on how to use the application.	----	----

Undo <action>	U <Ctrl><Z> (preferred) <Alt><Backspace>	U <Ctrl><Z> (preferred) <Alt><Backspace>
Reverses the most recently executed action.		

Unmark	--	----	--	----
Removes the annotation that an item has been selected.				
Unoccupy Workspace	U	----	--	----
Removes an application from the current workspace. If the application is only occupying one workspace, this option is displayed as unavailable.				
Update	--	----	--	----
Checks the status of a process and displays the updated information.				
Using Help	U	----	--	----
Provides information on how to use Help.				
View	--	----	V	----
Displays the contents of an item.				
Yes	--	----	Y	----
Indicates an affirmative response to a question and removes the window containing the question.				

APPENDIX D DEVELOPER NOTES

D.1 DESKTOP RESOURCES

This section lists the resource settings to implement the default DII configuration defined in section 7 of this style guide. Unless otherwise indicated, the default settings defined in the TED version of CDE are used.

Application Manager. Removing access to the Command Line Login feature in CDE limits users to launching only one instance of an application.

File Manager. The DtfilerestrictMode resource is set to True. This setting prevents users from accessing the Open Terminal option of the File menu in the Application and File Manager and from navigating above their own home directory. With regard to the latter, users desiring to share files will have to send them (e.g., by e-mail) to other users since they will be unable to see outside their own home directory.

D.2 COLOR SETS

Classification banner. The classification bar contains black text and uses the following colors to indicate security level:

Unclassified	Green	ForestGreen	34 139 34
Confidential	Blue	SkyBlue	135 206 235
Secret	Red	Firebrick2	238 44 44
Top Secret	Orange	DarkOrange	255 140 0

Color palettes. The following display elements are associated with the color set IDs used by CDE in the More Colors for Desktop (i.e., high color) setting in Style Manager:

Color Set ID	Display Element
-----------------	-----------------

- 1 Active window frame color, location cursor color, select color for toggle buttons
- 2 Inactive window frame color
- 3 Background color for workspace #1*
- 4 Background color for text entry areas and lists
- 5 Background color for application's main window areas, background color for workspace #2*
- 6 Background color for application's menu bar, menus, and dialog boxes, background color for workspace #3*
- 7 Background color for workspace #4*
- 8 Background for Front Panel and graphic in window icon

* This color is used when the Backdrop is set to "background."

Note: TED does not currently provide a common window background color across high, medium, and low color settings. This problem has been identified to TriTeal and will be corrected so that the software is consistent with the TED documentation.

Table D-1 lists the color name (where defined) and RGB value for each color set ID in the color palettes to be used in the DII configuration of CDE. The top set of colors in the table were selected from those included with CDE while the bottom set were created specifically for DII systems. The default color palette for DII systems is Gray75.

The letter in parentheses following the name of each palette indicates the foreground color (either (W)hite or (B)lack) used by CDE in the application's main window areas (i.e., color set ID 5). The contrast between foreground and background in these areas was computed using the equation provided in the DoD style guide and found to exceed the minimum contrast value of .35 recommended in that document. Because hardware platforms have different color maps, video boards, and monitors, developers may encounter color variations that diverge from the appearance intended when the color sets listed in table D-1 were developed.

Style Manager provides the ability to change application colors dynamically if the application is a Motif 1.1 or 1.2 client. Clients written with other toolkits cannot change color dynamically; color changes take effect when the client is restarted. The easiest way to use the dynamic colors provided by Style Manager is to remove any application color resources for background and foreground color. If the application currently specifies colors or is not in Motif, it is to use the colors indicated below. Otherwise, the application should not set colors and instead accept the defaults from the X Resource Manager.

Table D-1. Color names and RGB values for color palettes in the DII configuration.

<u>Color Set ID</u>	<u>Camouflage</u> (W)	<u>Charcoal</u> (W)	<u>CDE Default</u> (B)	<u>Northern Sky</u> (W)	<u>Sea Foam</u> (B)
1	172, 190, 180	135, 180, 176	237, 168, 112	164, 117, 145	170, 203, 219
2	110, 125, 118	118, 114, 121	153, 153, 153	27, 93, 108	167, 168, 167
3	144, 123, 123	105, 99, 99	137, 152, 170	12, 90, 135	125, 183, 187
4	130, 130, 155	137, 109, 109	104, 111, 130	0, 87, 122	151, 176, 178
5	142, 142, 119	130, 133, 133	198, 178, 168	65, 82, 92	175, 191, 186
6	127, 108, 108	99, 111, 109	73, 146, 167	75, 123, 130	197, 175, 164
7	142, 142, 133	145, 145, 145	183, 135, 141	39, 112, 139	152, 170, 165
8	78, 107, 101	175, 150, 150	147, 171, 191	61, 64, 124	200, 173, 160
	<u>Gray75</u> (B)	<u>SlateGray</u> (W)	<u>SteelBlue</u> (W)	<u>MistyRose2</u> (B)	
1	DarkSlateGray 47, 79, 79	Gray90 229, 229, 229	SlateGray2 185, 211, 238	DarkSlateBlue 72, 61, 139	
2	Gray75 191, 191, 191	SlateGray 112, 128, 144	SteelBlue 70, 130, 180	MistyRose2 238, 213, 211	
3	Gray60 153, 153, 153	Gray60 153, 153, 153	Gray60 153, 153, 153	Gray60 153, 153, 153	
4	Gray80 204, 204, 204	DarkSlateGray 47, 79, 79	SteelBlue4 54, 100, 139	Bisque 255, 228, 196	
5	Gray75 191, 191, 191	SlateGray 112, 128, 144	SteelBlue 70, 130, 180	MistyRose2 238, 213, 211	
6	Gray75 191, 191, 191	SlateGray 112, 128, 144	SteelBlue 70, 130, 180	MistyRose2 238, 213, 211	
7	Gray40 102, 102, 102	Gray40 102, 102, 102	Gray40 102, 102, 102	Gray40 102, 102, 102	
8	Gray40 102, 102, 102	Gray35 89, 89, 89	SteelBlue4 54, 100, 139	LightSkyBlue4 96, 123, 139	

The following backdrops are to be available in the DII configuration of CDE: Background, Concave, Convex, Foreground, No Backdrop, Paver, Sky Dark, and Sky Light.

D.3 FONT SETS

Applications in CDE or a previous version of Motif are to use the default CDE font aliases for the Hewlett-Packard and Sun platforms listed in appendix E. Style Manager provides the ability to change font size dynamically for those applications using CDE Motif; Style Manager can also be used to modify font size in non-CDE applications but users will have to restart the application for the change to take effect.

D.4 MOTIF RESOURCES

D.4.1 Resources for CDE and OSF Motif

The resource settings presented here focus on the set of widgets addressed in this style guide and include recommendations provided by Kobara in [Visual Design with OSF/Motif](#). The color set is the one identified as the DII default in the CDE implementation. Unless otherwise indicated, the default setting is assumed for resources related to shadow thickness, highlight thickness, traversal, highlight color, select color, arm color, trough color, top shadow color, bottom shadow color, margin height, and margin width.

CascadeButton

XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)
XmNmappingDelay:	Default (180)
XmNmarginHeight:	Default (2) (per Kobara)
XmNmarginWidth:	8 in a menu bar (per Kobara)

CommandBox

XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)

FileSelectionBox

XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)
XmNminimizeButtons:	Default (False)

Frame

XmNbackground:	Dynamic (DII default is Gray75)
XmNmarginHeight:	Not 0 (per Kobara)
XmNmarginWidth:	Not 0 (per Kobara)
XmNshadowThickness:	2 for Shadow In & Shadow Out frames, 1 for Etched In & Etched Out frames (per Kobara)
XmNshadowType:	XmSHADOW_ETCHED_IN

Label

XmNalignment:	XmALIGNMENT_BEGINNING for vertically
---------------	--------------------------------------

	stacked labels & labels with widgets to their left; XmALIGNMENT_END for labels with widgets to their right; otherwise XmALIGNMENT_CENTER (per Kobara)
XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)
XmNmarginHeight:	6 (per Kobara)
XmNmarginWidth:	Default (2) (per Kobara)

List

XmNbackground:	Dynamic (DII default is Gray80)
XmNforeground:	Dynamic (DII default is Black)
XmNlistMarginHeight:	2 (per Kobara)
XmNlistMarginWidth:	2 (per Kobara)
XmNlistSizePolicy:	XmCONSTANT (per Kobara)
XmNlistSpacing:	Default (0)
XmNscrollBarDisplayPolicy:	XmSTATIC (per Kobara)

MessageBox

XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)
XmNmessageAlignment:	Default (XmALIGNMENT_BEGINNING)
XmNminimizeButtons:	Default (False)

PanedWindow

XmNbackground:	Dynamic (DII default is Gray75)
XmNsashHeight:	6 (per Kobara)
XmNsashIndent:	Equal to the value by which vertical scroll bars are offset from the right edge of the paned window; -4 if no scroll bars are used (per Kobara)
XmNsashWidth:	Equal to the width of the scroll bars in the application; 12 if no scroll bars are used (per Kobara)
XmNspacing:	Default (8) (per Kobara)

PushButton

XmNalignment:	XmALIGNMENT_CENTER in a work area; XmALIGNMENT_LEFT in a menu
XmNbackground:	Dynamic (DII default is Gray75)
XmNfillOnArm:	True in a work area; False in a menu
XmNforeground:	Dynamic (DII default is Black)
XmNmarginHeight:	Default (2) (per Kobara)
XmNmarginWidth:	8 (per Kobara)

RowColumn

XmNadjustLast:	False (per Kobara)
XmNentryAlignment:	XmALIGNMENT_CENTER in a menu bar; XmALIGNMENT_BEGINNING in a menu
XmNentryBorder:	2 in a menu bar; 0 in a menu (per Kobara)

XmNorientation:	XmHORIZONTAL in a menu bar; XmVERTICAL in a menu
XmNspacing:	0 in a menu bar & menu (per Kobara)

Scale

XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)
XmNshowValue:	True

ScrollBar

XmNbackground:	Same as parent background color
XmNinitialDelay:	Default (250 ms)
XmNrepeatDelay:	Default (50 ms)
XmNscrollBarHeight:	16 for horizontal scrollbars (per Kobara)
XmNscrollBarWidth:	16 for vertical scrollbars (per Kobara)
XmNshowArrows:	Default (True)

ScrolledWindow

XmNscrollBarDisplayPolicy:	XmSTATIC (per Kobara)
XmNscrollBarPlacement:	XmBOTTOM_RIGHT (per Kobara)
XmNscrolledWindowMarginHeight:	Default (0) for scrolled list; 2 for scrolled text (per Kobara)
XmNscrolledWindowMarginWidth:	Default (0) for scrolled list; 2 for scrolled text (per Kobara)
XmNspacing:	Default (4) (per Kobara)

SelectionBox

XmNbackground:	Dynamic (DII default is Gray75)
XmNforeground:	Dynamic (DII default is Black)
XmNminimizeButtons:	Default (False)

Separator

XmNbackground:	Dynamic (DII default is Gray75)
XmNmargin:	Default (0) (per Kobara)
XmNseparatorType:	XmSHADOW_ETCHED_IN (per Kobara)

Text (Editable)

XmNautoShowCursorPosition:	Default (True)
XmNbackground:	Dynamic (DII default is Gray80)
XmNblinkRate:	Default (500 ms -- equals 2 Hz)
XmNcursorPositionVisible:	Default (True)
XmNeditable:	Default (True)
XmNforeground:	Dynamic (DII default is Black)
XmNwordWrap:	True (per Kobara)

Text (Noneditable)

XmNautoShowCursorPosition:	False
----------------------------	-------

XmNbackground:	Dynamic (DII default is Gray75)
XmNblinkRate:	0
XmNcursorPositionVisible:	False
XmNeditable:	False
XmNforeground:	Dynamic (DII default is Black)
XmNtraversalOn:	False

TextField (Editable)

XmNbackground:	Dynamic (DII default is Gray80)
XmNblinkRate:	Default (500)
XmNcursorPositionVisible:	Default (True)
XmNeditable:	Default (True)
XmNforeground:	Dynamic (DII default is Black)
XmNmarginHeight:	2 (per Kobara)
XmNmarginWidth:	Default (5)

TextField (Noneditable)

XmNbackground:	Dynamic (DII default is Gray75)
XmNblinkRate:	0
XmNcursorPositionVisible:	False
XmNeditable:	False
XmNforeground:	Dynamic (DII default is Black)
XmNhighlightThickness:	0
XmNmarginHeight:	2 (per Kobara)
XmNmarginWidth:	Default (5)
XmNtraversalOn:	False

ToggleButton

XmNbackground:	Dynamic (DII default is Gray75)
XmNfillOnSelect:	Default (True)
XmNforeground:	Dynamic (DII default is Black)
XmNindicatorOn:	Default (True)
XmNmarginHeight:	Default (2) (per Kobara)
XmNmarginWidth:	Default (2) (per Kobara)
XmNspacing:	Default (4) (per Kobara)
XmNvisibleWhenOff:	True in a work area & a menu

D.4.2 CDE-Specific Resources or Settings

Shadow thickness: CDE Motif defines a default shadow thickness of one pixel for components in application windows such as button shadows and focus highlight. Motif 1.2 applications use this resource value; other applications may not obtain this resource value and so may have a slightly different appearance than applications using CDE Motif or Motif 1.2.

CDE-specific widgets. The TED documentation provides the following recommendations concerning CDE-specific Motif resources:

FileSelectionBox

XmN.fullPathMode:	False
-------------------	-------

XmN.Dirttextfield:	"Enter Path Name or Folder Name:"
XmN.Filtertextfield:	"Filter"
XmN.Dirlist:	"Folder"
XmN.Contentslist:	"Files"
XmN.Filetext field:	"Enter File Name:"

The TED documentation recommends using the default values for resources related to SpinBox, ComboBox, MenuButton, and Editor.

D.5 APPLICATION ICONS

CDE levies several requirements with respect to the design of the icons in the application. First, because the desktop runs in both color and monochrome modes, icons need to be created in two formats: XPM for color and XBM for monochrome. Second, because the desktop accommodates low resolution (640 x 480 pixels), medium resolution (800 x 600 pixels), and high resolution (mega-pixel) displays, icons need to be created in three sizes: 16 x 16 pixels, 32 x 32 pixels, and 48 x 48 pixels. Table D-2 lists the minimum required icon set for the application.

Table D-2. Minimum required icon set.

Type of Icon	Color 16x16	Color 32x32	Color 48x48	Mono 16x16	Mono 32x32	Mono 48x48
Application Icon	Required	Required	Required	Required	Required	Required
Document or File Icon	Required	Required		Required	Required	
Application Container Icon	Required	Required		Required	Required	
Minimized Windows			Required			Required

Icon names are to be no more than seven characters and include two suffixes, one for size and one for format, in the form:

iconname.size.format

Size is a single letter indicating the standard size of the icon: l for large (48 x 48 pixels), m for medium (32 x 32 pixels), s for small (24 x 24 pixels), and t for tiny (16 x 16 pixels). (Icons that are 24 x 24 are used for internal application graphics (e.g., in toolbars) and are not part of the standard set of desktop icons.) Format is pm for X pixmaps (the color icon format) or bm for X bitmaps (the monochrome icon format).

The Icon Editor in CDE is used to create desktop icons. This editor provides a palette of eight static grays, eight static colors, five dynamic colors (e.g., foreground, background), and a transparent "color" that allows the background to show through. The application uses this default palette and design icons primarily in grays, with other colors used sparingly. When designing the icon image, 16 x 16 and 32 x 32 icons are left-aligned within the bounding box for the image, with any empty bits to the right, and 48 x 48 icons are centered in the bounding box.

D.6 NOTES ON MOTIF IMPLEMENTATION

Developer Notes on Motif implementation will be provided in future versions of this document.

D.7 NOTES ON WINDOWS IMPLEMENTATION

Fonts. Segments shall use TrueType fonts to take advantage of the increased performance, flexibility, and WYSIWYG screen-to-printer characteristics. Custom application-specific fonts shall be avoided in favor of using industry standard fonts wherever possible. (from the DII COE Integration and Runtime Environment Specification)

Additional Developer Notes on Windows implementation will be provided in future versions of this document.

APPENDIX E FONT LIST

This appendix lists the fonts for which font aliases are available in CDE for the Hewlett-Packard (i.e., HP 9.*) and Sun (i.e., Solaris 2.*) platforms. Developers can contact DISA to obtain the complete list of font aliases for each platform.

Hewlett-Packard Application Fonts

"-adobe-courier-bold-o-normal--11-80-100-100-m-60-iso8859-1"
"-adobe-courier-bold-o-normal--14-100-100-100-m-90-iso8859-1"
"-adobe-courier-bold-o-normal--17-120-100-100-m-100-iso8859-1"
"-adobe-courier-bold-o-normal--20-140-100-100-m-110-iso8859-1"
"-adobe-courier-bold-o-normal--25-180-100-100-m-150-iso8859-1"
"-adobe-courier-bold-o-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-bold-o-normal--8-80-75-75-m-50-iso8859-1"
"-adobe-courier-bold-o-normal--10-100-75-75-m-60-iso8859-1"
"-adobe-courier-bold-o-normal--12-120-75-75-m-70-iso8859-1"
"-adobe-courier-bold-o-normal--14-140-75-75-m-90-iso8859-1"
"-adobe-courier-bold-o-normal--18-180-75-75-m-110-iso8859-1"
"-adobe-courier-bold-o-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-helvetica-bold-o-normal--11-80-100-100-p-60-iso8859-1"
"-adobe-helvetica-bold-o-normal--14-100-100-100-p-82-iso8859-1"
"-adobe-helvetica-bold-o-normal--17-120-100-100-p-92-iso8859-1"
"-adobe-helvetica-bold-o-normal--20-140-100-100-p-103-iso8859-1"
"-adobe-helvetica-bold-o-normal--25-180-100-100-p-138-iso8859-1"
"-adobe-helvetica-bold-o-normal--34-240-100-100-p-182-iso8859-1"

"-adobe-new century schoolbook-bold-i-normal--11-80-100-100-p-66-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--14-100-100-100-p-88-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--17-120-100-100-p-99-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--20-140-100-100-p-111-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--25-180-100-100-p-148-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--34-240-100-100-p-193-iso8859-1"

"-adobe-helvetica-bold-o-normal--8-80-75-75-p-50-iso8859-1"
"-adobe-helvetica-bold-o-normal--10-100-75-75-p-60-iso8859-1"
"-adobe-helvetica-bold-o-normal--12-120-75-75-p-69-iso8859-1"
"-adobe-helvetica-bold-o-normal--14-140-75-75-p-82-iso8859-1"
"-adobe-helvetica-bold-o-normal--18-180-75-75-p-104-iso8859-1"
"-adobe-helvetica-bold-o-normal--24-240-75-75-p-138-iso8859-1"

"-adobe-new century schoolbook-bold-i-normal--8-80-75-75-p-56-iso8859-1"

"-adobe-new century schoolbook-bold-i-normal--10-100-75-75-p-66-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--12-120-75-75-p-76-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--14-140-75-75-p-88-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--18-180-75-75-p-111-iso8859-1"
"-adobe-new century schoolbook-bold-i-normal--24-240-75-75-p-148-iso8859-1"

"-adobe-courier-bold-r-normal--11-80-100-100-m-60-iso8859-1"
"-adobe-courier-bold-r-normal--14-100-100-100-m-90-iso8859-1"
"-adobe-courier-bold-r-normal--17-120-100-100-m-100-iso8859-1"
"-adobe-courier-bold-r-normal--20-140-100-100-m-110-iso8859-1"
"-adobe-courier-bold-r-normal--25-180-100-100-m-150-iso8859-1"
"-adobe-courier-bold-r-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-bold-r-normal--8-80-75-75-m-50-iso8859-1"
"-adobe-courier-bold-r-normal--10-100-75-75-m-60-iso8859-1"
"-adobe-courier-bold-r-normal--12-120-75-75-m-70-iso8859-1"
"-adobe-courier-bold-r-normal--14-140-75-75-m-90-iso8859-1"
"-adobe-courier-bold-r-normal--18-180-75-75-m-110-iso8859-1"
"-adobe-courier-bold-r-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-helvetica-bold-r-normal--11-80-100-100-p-60-iso8859-1"
"-adobe-helvetica-bold-r-normal--14-100-100-100-p-82-iso8859-1"
"-adobe-helvetica-bold-r-normal--17-120-100-100-p-92-iso8859-1"
"-adobe-helvetica-bold-r-normal--20-140-100-100-p-105-iso8859-1"
"-adobe-helvetica-bold-r-normal--25-180-100-100-p-138-iso8859-1"
"-adobe-helvetica-bold-r-normal--34-240-100-100-p-182-iso8859-1"

"-adobe-new century schoolbook-bold-r-normal--11-80-100-100-p-66-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--14-100-100-100-p-87-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--17-120-100-100-p-99-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--20-140-100-100-p-113-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--25-180-100-100-p-149-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--34-240-100-100-p-193-iso8859-1"

"-adobe-helvetica-bold-r-normal--8-80-75-75-p-50-iso8859-1"
"-adobe-helvetica-bold-r-normal--10-100-75-75-p-60-iso8859-1"
"-adobe-helvetica-bold-r-normal--12-120-75-75-p-70-iso8859-1"
"-adobe-helvetica-bold-r-normal--14-140-75-75-p-82-iso8859-1"
"-adobe-helvetica-bold-r-normal--18-180-75-75-p-103-iso8859-1"
"-adobe-helvetica-bold-r-normal--24-240-75-75-p-138-iso8859-1"

"-adobe-new century schoolbook-bold-r-normal--8-80-75-75-p-56-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--10-100-75-75-p-66-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--12-120-75-75-p-77-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--14-140-75-75-p-87-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--18-180-75-75-p-113-iso8859-1"
"-adobe-new century schoolbook-bold-r-normal--24-240-75-75-p-149-iso8859-1"

"-adobe-courier-medium-o-normal--11-80-100-100-m-60-iso8859-1"
"-adobe-courier-medium-o-normal--14-100-100-100-m-90-iso8859-1"
"-adobe-courier-medium-o-normal--17-120-100-100-m-100-iso8859-1"
"-adobe-courier-medium-o-normal--20-140-100-100-m-110-iso8859-1"
"-adobe-courier-medium-o-normal--25-180-100-100-m-150-iso8859-1"
"-adobe-courier-medium-o-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-medium-o-normal--8-80-75-75-m-50-iso8859-1"
"-adobe-courier-medium-o-normal--10-100-75-75-m-60-iso8859-1"
"-adobe-courier-medium-o-normal--12-120-75-75-m-70-iso8859-1"
"-adobe-courier-medium-o-normal--14-140-75-75-m-90-iso8859-1"
"-adobe-courier-medium-o-normal--18-180-75-75-m-110-iso8859-1"
"-adobe-courier-medium-o-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-helvetica-medium-o-normal--11-80-100-100-p-57-iso8859-1"
"-adobe-helvetica-medium-o-normal--14-100-100-100-p-78-iso8859-1"
"-adobe-helvetica-medium-o-normal--17-120-100-100-p-88-iso8859-1"
"-adobe-helvetica-medium-o-normal--20-140-100-100-p-98-iso8859-1"
"-adobe-helvetica-medium-o-normal--25-180-100-100-p-130-iso8859-1"
"-adobe-helvetica-medium-o-normal--34-240-100-100-p-176-iso8859-1"

"-adobe-new century schoolbook-medium-i-normal--11-80-100-100-p-60-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--14-100-100-100-p-81-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--17-120-100-100-p-92-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--20-140-100-100-p-104-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--25-180-100-100-p-136-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--34-240-100-100-p-182-iso8859-1"

"-adobe-helvetica-medium-o-normal--8-80-75-75-p-47-iso8859-1"
"-adobe-helvetica-medium-o-normal--10-100-75-75-p-57-iso8859-1"
"-adobe-helvetica-medium-o-normal--12-120-75-75-p-67-iso8859-1"
"-adobe-helvetica-medium-o-normal--14-140-75-75-p-78-iso8859-1"
"-adobe-helvetica-medium-o-normal--18-180-75-75-p-98-iso8859-1"
"-adobe-helvetica-medium-o-normal--24-240-75-75-p-130-iso8859-1"

"-adobe-new century schoolbook-medium-i-normal--8-80-75-75-p-50-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--10-100-75-75-p-60-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--12-120-75-75-p-70-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--14-140-75-75-p-81-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--18-180-75-75-p-104-iso8859-1"
"-adobe-new century schoolbook-medium-i-normal--24-240-75-75-p-136-iso8859-1"

"-adobe-courier-medium-r-normal--11-80-100-100-m-60-iso8859-1"
"-adobe-courier-medium-r-normal--14-100-100-100-m-90-iso8859-1"
"-adobe-courier-medium-r-normal--17-120-100-100-m-100-iso8859-1"
"-adobe-courier-medium-r-normal--20-140-100-100-m-110-iso8859-1"
"-adobe-courier-medium-r-normal--25-180-100-100-m-150-iso8859-1"
"-adobe-courier-medium-r-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-medium-r-normal--8-80-75-75-m-50-iso8859-1"
"-adobe-courier-medium-r-normal--10-100-75-75-m-60-iso8859-1"
"-adobe-courier-medium-r-normal--12-120-75-75-m-70-iso8859-1"
"-adobe-courier-medium-r-normal--14-140-75-75-m-90-iso8859-1"
"-adobe-courier-medium-r-normal--18-180-75-75-m-110-iso8859-1"
"-adobe-courier-medium-r-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-symbol-medium-r-normal--11-80-100-100-p-61-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--14-100-100-100-p-85-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--17-120-100-100-p-95-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--20-140-100-100-p-107-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--25-180-100-100-p-142-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--34-240-100-100-p-191-adobe-fontspecific"

"-adobe-helvetica-medium-r-normal--11-80-100-100-p-56-iso8859-1"
"-adobe-helvetica-medium-r-normal--14-100-100-100-p-76-iso8859-1"
"-adobe-helvetica-medium-r-normal--17-120-100-100-p-88-iso8859-1"
"-adobe-helvetica-medium-r-normal--20-140-100-100-p-100-iso8859-1"
"-adobe-helvetica-medium-r-normal--25-180-100-100-p-130-iso8859-1"
"-adobe-helvetica-medium-r-normal--34-240-100-100-p-176-iso8859-1"

"-adobe-new century schoolbook-medium-r-normal--11-80-100-100-p-60-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--14-100-100-100-p-82-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--17-120-100-100-p-91-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--20-140-100-100-p-103-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--25-180-100-100-p-136-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--34-240-100-100-p-181-iso8859-1"

"-adobe-symbol-medium-r-normal--8-80-75-75-p-51-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--10-100-75-75-p-61-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--12-120-75-75-p-74-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--14-140-75-75-p-85-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--18-180-75-75-p-107-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--24-240-75-75-p-142-adobe-fontspecific"

"-adobe-helvetica-medium-r-normal--8-80-75-75-p-46-iso8859-1"
"-adobe-helvetica-medium-r-normal--10-100-75-75-p-56-iso8859-1"
"-adobe-helvetica-medium-r-normal--12-120-75-75-p-67-iso8859-1"
"-adobe-helvetica-medium-r-normal--14-140-75-75-p-77-iso8859-1"
"-adobe-helvetica-medium-r-normal--18-180-75-75-p-98-iso8859-1"
"-adobe-helvetica-medium-r-normal--24-240-75-75-p-130-iso8859-1"

"-adobe-new century schoolbook-medium-r-normal--8-80-75-75-p-50-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--10-100-75-75-p-60-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--12-120-75-75-p-70-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--14-140-75-75-p-82-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--18-180-75-75-p-103-iso8859-1"
"-adobe-new century schoolbook-medium-r-normal--24-240-75-75-p-137-iso8859-1"

Hewlett-Packard Interface (System) Fonts

"-hp-hp system-medium-r-normal-sans-10-100-72-72-p-61-iso8859-1"
"-hp-hp system-medium-r-normal-sans-11-110-72-72-p-64-iso8859-1"
"-hp-hp system-medium-r-normal-sans-12-120-72-72-p-73-iso8859-1"
"-hp-hp system-medium-r-normal-sans-13-130-72-72-p-87-iso8859-1"
"-hp-hp system-medium-r-normal-sans-15-150-72-72-p-100-iso8859-1"
"-hp-hp system-medium-r-normal-sans-18-180-72-72-p-114-iso8859-1"
"-hp-hp system-medium-r-normal-sans-21-210-72-72-p-123-iso8859-1"

Hewlett-Packard Interface (User) Fonts

"-hp-hp user-medium-r-normal-serif-10-100-72-72-m-60-iso8859-1"
"-hp-hp user-medium-r-normal-serif-11-110-72-72-m-60-iso8859-1"
"-hp-hp user-medium-r-normal-serif-12-120-72-72-m-80-iso8859-1"
"-hp-hp user-medium-r-normal-serif-13-130-72-72-m-90-iso8859-1"
"-hp-hp user-medium-r-normal-serif-15-150-72-72-m-110-iso8859-1"
"-hp-hp user-medium-r-normal-serif-18-180-72-72-m-120-iso8859-1"

"-hp-hp user-medium-r-normal-serif-21-210-72-72-m-140-iso8859-1"

"-hp-hp user-bold-r-normal-serif-10-100-72-72-m-60-iso8859-1"

"-hp-hp user-bold-r-normal-serif-11-110-72-72-m-60-iso8859-1"

"-hp-hp user-bold-r-normal-serif-12-120-72-72-m-80-iso8859-1"

"-hp-hp user-bold-r-normal-serif-13-130-72-72-m-90-iso8859-1"

"-hp-hp user-bold-r-normal-serif-15-150-72-72-m-110-iso8859-1"

"-hp-hp user-bold-r-normal-serif-18-180-72-72-m-120-iso8859-1"

"-hp-hp user-bold-r-normal-serif-21-210-72-72-m-140-iso8859-1"

Sun Application Fonts

"-adobe-courier-bold-o-normal--11-80-100-100-m-60-iso8859-1"

"-adobe-courier-bold-o-normal--14-100-100-100-m-90-iso8859-1"

"-adobe-courier-bold-o-normal--17-120-100-100-m-100-iso8859-1"

"-adobe-courier-bold-o-normal--20-140-100-100-m-110-iso8859-1"

"-adobe-courier-bold-o-normal--25-180-100-100-m-150-iso8859-1"

"-adobe-courier-bold-o-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-bold-o-normal--8-80-75-75-m-50-iso8859-1"

"-adobe-courier-bold-o-normal--10-100-75-75-m-60-iso8859-1"

"-adobe-courier-bold-o-normal--12-120-75-75-m-70-iso8859-1"

"-adobe-courier-bold-o-normal--14-140-75-75-m-90-iso8859-1"

"-adobe-courier-bold-o-normal--18-180-75-75-m-110-iso8859-1"

"-adobe-courier-bold-o-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-helvetica-bold-o-normal--11-80-100-100-p-60-iso8859-1"

"-adobe-helvetica-bold-o-normal--14-100-100-100-p-82-iso8859-1"

"-adobe-helvetica-bold-o-normal--17-120-100-100-p-92-iso8859-1"

"-adobe-helvetica-bold-o-normal--20-140-100-100-p-103-iso8859-1"

"-adobe-helvetica-bold-o-normal--25-180-100-100-p-138-iso8859-1"

"-adobe-helvetica-bold-o-normal--34-240-100-100-p-182-iso8859-1"

"-adobe-times-bold-i-normal--11-80-100-100-p-57-iso8859-1"

"-adobe-times-bold-i-normal--14-100-100-100-p-77-iso8859-1"

"-adobe-times-bold-i-normal--17-120-100-100-p-86-iso8859-1"

"-adobe-times-bold-i-normal--20-140-100-100-p-98-iso8859-1"

"-adobe-times-bold-i-normal--25-180-100-100-p-128-iso8859-1"

"-adobe-times-bold-i-normal--34-240-100-100-p-170-iso8859-1"

"-adobe-helvetica-bold-o-normal--8-80-75-75-p-50-iso8859-1"

"-adobe-helvetica-bold-o-normal--10-100-75-75-p-60-iso8859-1"

"-adobe-helvetica-bold-o-normal--12-120-75-75-p-69-iso8859-1"

"-adobe-helvetica-bold-o-normal--14-140-75-75-p-82-iso8859-1"

"-adobe-helvetica-bold-o-normal--18-180-75-75-p-104-iso8859-1"

"-adobe-helvetica-bold-o-normal--24-240-75-75-p-138-iso8859-1"

"-adobe-times-bold-i-normal--8-80-75-75-p-47-iso8859-1"

"-adobe-times-bold-i-normal--10-100-75-75-p-57-iso8859-1"

"-adobe-times-bold-i-normal--12-120-75-75-p-68-iso8859-1"

"-adobe-times-bold-i-normal--14-140-75-75-p-77-iso8859-1"

"-adobe-times-bold-i-normal--18-180-75-75-p-98-iso8859-1"

"-adobe-times-bold-i-normal--24-240-75-75-p-128-iso8859-1"

"-adobe-courier-bold-r-normal--11-80-100-100-m-60-iso8859-1"
"-adobe-courier-bold-r-normal--14-100-100-100-m-90-iso8859-1"
"-adobe-courier-bold-r-normal--17-120-100-100-m-100-iso8859-1"
"-adobe-courier-bold-r-normal--20-140-100-100-m-110-iso8859-1"
"-adobe-courier-bold-r-normal--25-180-100-100-m-150-iso8859-1"
"-adobe-courier-bold-r-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-bold-r-normal--8-80-75-75-m-50-iso8859-1"
"-adobe-courier-bold-r-normal--10-100-75-75-m-60-iso8859-1"
"-adobe-courier-bold-r-normal--12-120-75-75-m-70-iso8859-1"
"-adobe-courier-bold-r-normal--14-140-75-75-m-90-iso8859-1"
"-adobe-courier-bold-r-normal--18-180-75-75-m-110-iso8859-1"
"-adobe-courier-bold-r-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-helvetica-bold-r-normal--11-80-100-100-p-60-iso8859-1"
"-adobe-helvetica-bold-r-normal--14-100-100-100-p-82-iso8859-1"
"-adobe-helvetica-bold-r-normal--17-120-100-100-p-92-iso8859-1"
"-adobe-helvetica-bold-r-normal--20-140-100-100-p-105-iso8859-1"
"-adobe-helvetica-bold-r-normal--25-180-100-100-p-138-iso8859-1"
"-adobe-helvetica-bold-r-normal--34-240-100-100-p-182-iso8859-1"

"-adobe-times-bold-r-normal--11-80-100-100-p-57-iso8859-1"
"-adobe-times-bold-r-normal--14-100-100-100-p-76-iso8859-1"
"-adobe-times-bold-r-normal--17-120-100-100-p-88-iso8859-1"
"-adobe-times-bold-r-normal--20-140-100-100-p-100-iso8859-1"
"-adobe-times-bold-r-normal--25-180-100-100-p-132-iso8859-1"
"-adobe-times-bold-r-normal--34-240-100-100-p-177-iso8859-1"

"-adobe-helvetica-bold-r-normal--8-80-75-75-p-50-iso8859-1"
"-adobe-helvetica-bold-r-normal--10-100-75-75-p-60-iso8859-1"
"-adobe-helvetica-bold-r-normal--12-120-75-75-p-70-iso8859-1"
"-adobe-helvetica-bold-r-normal--14-140-75-75-p-82-iso8859-1"
"-adobe-helvetica-bold-r-normal--18-180-75-75-p-103-iso8859-1"
"-adobe-helvetica-bold-r-normal--24-240-75-75-p-138-iso8859-1"

"-adobe-times-bold-r-normal--8-80-75-75-p-47-iso8859-1"
"-adobe-times-bold-r-normal--10-100-75-75-p-57-iso8859-1"
"-adobe-times-bold-r-normal--12-120-75-75-p-67-iso8859-1"
"-adobe-times-bold-r-normal--14-140-75-75-p-77-iso8859-1"
"-adobe-times-bold-r-normal--18-180-75-75-p-99-iso8859-1"
"-adobe-times-bold-r-normal--24-240-75-75-p-132-iso8859-1"

"-adobe-courier-medium-o-normal--11-80-100-100-m-60-iso8859-1"
"-adobe-courier-medium-o-normal--14-100-100-100-m-90-iso8859-1"
"-adobe-courier-medium-o-normal--17-120-100-100-m-100-iso8859-1"
"-adobe-courier-medium-o-normal--20-140-100-100-m-110-iso8859-1"
"-adobe-courier-medium-o-normal--25-180-100-100-m-150-iso8859-1"
"-adobe-courier-medium-o-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-medium-o-normal--8-80-75-75-m-50-iso8859-1"
"-adobe-courier-medium-o-normal--10-100-75-75-m-60-iso8859-1"
"-adobe-courier-medium-o-normal--12-120-75-75-m-70-iso8859-1"
"-adobe-courier-medium-o-normal--14-140-75-75-m-90-iso8859-1"
"-adobe-courier-medium-o-normal--18-180-75-75-m-110-iso8859-1"

"-adobe-courier-medium-o-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-helvetica-medium-o-normal--11-80-100-100-p-57-iso8859-1"

"-adobe-helvetica-medium-o-normal--14-100-100-100-p-78-iso8859-1"

"-adobe-helvetica-medium-o-normal--17-120-100-100-p-88-iso8859-1"

"-adobe-helvetica-medium-o-normal--20-140-100-100-p-98-iso8859-1"

"-adobe-helvetica-medium-o-normal--25-180-100-100-p-130-iso8859-1"

"-adobe-helvetica-medium-o-normal--34-240-100-100-p-176-iso8859-1"

"-adobe-times-medium-i-normal--11-80-100-100-p-52-iso8859-1"

"-adobe-times-medium-i-normal--14-100-100-100-p-73-iso8859-1"

"-adobe-times-medium-i-normal--17-120-100-100-p-84-iso8859-1"

"-adobe-times-medium-i-normal--20-140-100-100-p-94-iso8859-1"

"-adobe-times-medium-i-normal--25-180-100-100-p-125-iso8859-1"

"-adobe-times-medium-i-normal--34-240-100-100-p-168-iso8859-1"

"-adobe-helvetica-medium-o-normal--8-80-75-75-p-47-iso8859-1"

"-adobe-helvetica-medium-o-normal--10-100-75-75-p-57-iso8859-1"

"-adobe-helvetica-medium-o-normal--12-120-75-75-p-67-iso8859-1"

"-adobe-helvetica-medium-o-normal--14-140-75-75-p-78-iso8859-1"

"-adobe-helvetica-medium-o-normal--18-180-75-75-p-98-iso8859-1"

"-adobe-helvetica-medium-o-normal--24-240-75-75-p-130-iso8859-1"

"-adobe-times-medium-i-normal--8-80-75-75-p-42-iso8859-1"

"-adobe-times-medium-i-normal--10-100-75-75-p-52-iso8859-1"

"-adobe-times-medium-i-normal--12-120-75-75-p-63-iso8859-1"

"-adobe-times-medium-i-normal--14-140-75-75-p-73-iso8859-1"

"-adobe-times-medium-i-normal--18-180-75-75-p-94-iso8859-1"

"-adobe-times-medium-i-normal--24-240-75-75-p-125-iso8859-1"

"-adobe-courier-medium-r-normal--11-80-100-100-m-60-iso8859-1"

"-adobe-courier-medium-r-normal--14-100-100-100-m-90-iso8859-1"

"-adobe-courier-medium-r-normal--17-120-100-100-m-100-iso8859-1"

"-adobe-courier-medium-r-normal--20-140-100-100-m-110-iso8859-1"

"-adobe-courier-medium-r-normal--25-180-100-100-m-150-iso8859-1"

"-adobe-courier-medium-r-normal--34-240-100-100-m-200-iso8859-1"

"-adobe-courier-medium-r-normal--8-80-75-75-m-50-iso8859-1"

"-adobe-courier-medium-r-normal--10-100-75-75-m-60-iso8859-1"

"-adobe-courier-medium-r-normal--12-120-75-75-m-70-iso8859-1"

"-adobe-courier-medium-r-normal--14-140-75-75-m-90-iso8859-1"

"-adobe-courier-medium-r-normal--18-180-75-75-m-110-iso8859-1"

"-adobe-courier-medium-r-normal--24-240-75-75-m-150-iso8859-1"

"-adobe-symbol-medium-r-normal--11-80-100-100-p-61-adobe-fontspecific"

"-adobe-symbol-medium-r-normal--14-100-100-100-p-85-adobe-fontspecific"

"-adobe-symbol-medium-r-normal--17-120-100-100-p-95-adobe-fontspecific"

"-adobe-symbol-medium-r-normal--20-140-100-100-p-107-adobe-fontspecific"

"-adobe-symbol-medium-r-normal--25-180-100-100-p-142-adobe-fontspecific"

"-adobe-symbol-medium-r-normal--34-240-100-100-p-191-adobe-fontspecific"

"-adobe-helvetica-medium-r-normal--11-80-100-100-p-56-iso8859-1"

"-adobe-helvetica-medium-r-normal--14-100-100-100-p-76-iso8859-1"

"-adobe-helvetica-medium-r-normal--17-120-100-100-p-88-iso8859-1"

"-adobe-helvetica-medium-r-normal--20-140-100-100-p-100-iso8859-1"
"-adobe-helvetica-medium-r-normal--25-180-100-100-p-130-iso8859-1"
"-adobe-helvetica-medium-r-normal--34-240-100-100-p-176-iso8859-1"

"-adobe-times-medium-r-normal--11-80-100-100-p-54-iso8859-1"
"-adobe-times-medium-r-normal--14-100-100-100-p-74-iso8859-1"
"-adobe-times-medium-r-normal--17-120-100-100-p-84-iso8859-1"
"-adobe-times-medium-r-normal--20-140-100-100-p-96-iso8859-1"
"-adobe-times-medium-r-normal--25-180-100-100-p-125-iso8859-1"
"-adobe-times-medium-r-normal--34-240-100-100-p-170-iso8859-1"

"-adobe-symbol-medium-r-normal--8-80-75-75-p-51-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--10-100-75-75-p-61-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--12-120-75-75-p-74-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--14-140-75-75-p-85-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--18-180-75-75-p-107-adobe-fontspecific"
"-adobe-symbol-medium-r-normal--24-240-75-75-p-142-adobe-fontspecific"

"-adobe-helvetica-medium-r-normal--8-80-75-75-p-46-iso8859-1"
"-adobe-helvetica-medium-r-normal--10-100-75-75-p-56-iso8859-1"
"-adobe-helvetica-medium-r-normal--12-120-75-75-p-67-iso8859-1"
"-adobe-helvetica-medium-r-normal--14-140-75-75-p-77-iso8859-1"
"-adobe-helvetica-medium-r-normal--18-180-75-75-p-98-iso8859-1"
"-adobe-helvetica-medium-r-normal--24-240-75-75-p-130-iso8859-1"

"-adobe-times-medium-r-normal--8-80-75-75-p-44-iso8859-1"
"-adobe-times-medium-r-normal--10-100-75-75-p-54-iso8859-1"
"-adobe-times-medium-r-normal--12-120-75-75-p-64-iso8859-1"
"-adobe-times-medium-r-normal--14-140-75-75-p-74-iso8859-1"
"-adobe-times-medium-r-normal--18-180-75-75-p-94-iso8859-1"
"-adobe-times-medium-r-normal--24-240-75-75-p-124-iso8859-1"

Sun Interface (System) Fonts

"-b&h-lucida sans-medium-r-normal-sans-10-100-72-72-p-59-iso8859-1"
"-b&h-lucida sans-medium-r-normal-sans-12-120-72-72-p-71-iso8859-1"
"-b&h-lucida sans-medium-r-normal-sans-14-140-72-72-p-82-iso8859-1"
"-b&h-lucida-medium-r-normal-sans-17-120-100-100-p-96-iso8859-1"
"-b&h-lucida sans-medium-r-normal-sans-18-180-72-72-p-106-iso8859-1"
"-b&h-lucida-medium-r-normal-sans-20-140-100-100-p-114-iso8859-1"
"-b&h-lucida-medium-r-normal-sans-24-240-75-75-p-136-iso8859-1"

Sun Interface (User) Fonts

"-b&h-lucida sans typewriter-bold-r-normal-sans-8-80-72-72-m-50-iso8859-1"
"-b&h-lucida sans typewriter-bold-r-normal-sans-10-100-72-72-m-60-iso8859-1"
"-b&h-lucida sans typewriter-bold-r-normal-sans-12-120-72-72-m-70-iso8859-1"
"-b&h-lucida sans typewriter-bold-r-normal-sans-14-140-72-72-m-90-iso8859-1"

"-b&h-lucida typewriter-bold-r-normal-sans-17-120-100-100-m-100-iso8859-1"
"-b&h-lucida typewriter-bold-r-normal-sans-20-140-100-100-m-120-iso8859-1"
"-b&h-lucida typewriter-bold-r-normal-sans-24-240-75-75-m-140-iso8859-1"

"-b&h-lucida sans typewriter-medium-r-normal-sans-8-80-72-72-m-50-iso8859-1"

"-b&h-lucida sans typewriter-medium-r-normal-sans-10-100-72-72-m-60-iso8859-1"
 "-b&h-lucida sans typewriter-medium-r-normal-sans-12-120-72-72-m-70-iso8859-1"
 "-b&h-lucida sans typewriter-medium-r-normal-sans-14-140-72-72-m-90-iso8859-1"

"-b&h-lucidatypewriter-medium-r-normal-sans-17-120-100-100-m-100-iso8859-1"
 "-b&h-lucidatypewriter-medium-r-normal-sans-20-140-100-100-m-120-iso8859-1"
 "-b&h-lucidatypewriter-medium-r-normal-sans-24-240-75-75-m-140-iso8859-1"

APPENDIX F

ACRONYMS AND ABBREVIATIONS

ASCII	American Standard Code for Information Exchange
CDE	Common Desktop Environment
COE	Common Operating Environment
COTS	Commercial Off-the-Shelf
CTM	Centralized Task Management
DIA	Defense Intelligence Agency
DII	Defense Information Infrastructure
DISA	Defense Information Systems Agency
DoD	Department of Defense
DTM	Distributed Task Management
GCCS	Global Command and Control System
GCSS	Global Command Support System
GUI	Graphical User Interface
HP	Hewlett Packard
HTML	HyperText Markup Language
I&RTS	Integration and Runtime Specification
ISO	International Organization for Standardization
MIL-HDBK	Military Handbook
MIL-STD	Military Standard
NATO	North Atlantic Treaty Organization
PC	Personal Computer
POSIX	Portable Operating System Interface for Computer Environments
TED	TriTeal Enterprise Desktop
UIS	User Interface Specifications

APPENDIX G

MOTIF AND WINDOWS TERMINOLOGY

<u>DII Term</u>	<u>Motif Term</u>	<u>Windows Term</u>
Accelerator	Accelerator	Shortcut key
Add mode	Add mode	Explicit selection

Autorepeat	Autorepeat	-----
Autoscrolling	Autoscrolling	Automatic scrolling
Cancel activation	Cancel activation	-----
Check button	Check button	Check box
Combo box	Combo box	Combo box
Dialog window	Dialog box	Dialog box
Discontiguous selection	Discontiguous selection	Disjoint selection
Document window	-----	Document window
Drag transfer	Drag transfer	Drag and drop
Drop-down combo box	Combo box	Drop-down combo box
Drop-down list box	-----	Drop-down list box
Expert activation	Expert activation	-----
Extend mode	-----	Extend mode
Frame	Panel	Group box
GoSub button	-----	GoSub button
GoTo button	-----	GoTo button
Hotspot	Hotspot	Hotspot
Input focus	Input focus	Input focus
Label	Label	Control label
List box	List	List box
Location cursor	Location cursor	Active control indicator
Margin selection	-----	Margin selection
Maximize button	Maximize button	Maximize button
Menu bar	Menu bar	Menu bar
Menu window	Menu window	-----
Message bar	Message area	Message bar
Message window	Message box	Message box
Minimize button	Minimize button	Minimize button
Mnemonic	Mnemonic	Mnemonic
Multiple selection list box	-----	Multiple selection list box
Noneditable text field	-----	Static text field
Normal mode	Normal mode	Implicit selection
Option button	Option button	-----
Option menu	Option menu	-----
Pane	Pane	Pane
Pointer	Pointer	Cursor
Pointing device	Pointing device	Pointing device
Pop-up menu	Pop-up menu	Pop-up menu
Previewing	Previewing	-----
Primary transfer	Primary transfer	-----
Primary window	Primary window	Primary window
Pull-down menu	Pull-down menu	Drop-down menu

Push button	Push button	Command button
Quick transfer	Quick transfer	-----
Radio button	Radio button	Option button
Range selection	Range selection	Contiguous selection
Read-only pop-up text box	-----	Read-only pop-up text box
Restore button	-----	Restore button
Scale	Scale	Slider
Scroll bar	Scroll bar	Scroll bar
Scrolling	Scrolling	Explicit scrolling
Secondary window	Secondary window	Secondary window
Separator	Separator	Separator
Spin button	Spin box	Spin box
Split bar	-----	Split bar
Split box	-----	Split box
Status bar	Status bar	Status bar
Submenu	Cascading menu	Cascading menu
Tear-off button	Tear-off button	-----
Tear-off menu	Tear-off menu	-----
Text cursor	Text cursor	Insertion point
Text field	Text field	Text box
Title bar	Title bar	Title bar
Tool bar	Tool bar	Control bar
Window	Window	Window
Window frame	Window border	Window frame
Window menu	Window menu	Control menu
Window menu button	Window menu button	Control menu box

APPENDIX H

STYLE REQUIREMENTS FOR COE COMPLIANCE

This appendix defines the specific requirements that must be satisfied in order to be considered compliant with each of the style-related items in the COE compliance checklist published in the DII I&RTS. The requirements are mapped to specific sections of the DII style guide and, as in the I&RTS, are intended to provide a migration path for existing software to become fully COE compliant in the style area. New software is expected to be developed in accordance with DII requirements and to be fully compliant with all style guide specifications.

Note: The wording below reflects changes to the compliance checklist to be published in version 2.1 of the I&RTS.

Level 1 Compliance

The application complies with the style of the native GUI.

- ___ (a) The application meets the minimum style requirements listed in section 1.5.2.3 of the DII Style Guide.
- ___ (b) If X-based, the application uses Motif.
- ___ (c) If Windows-based, the application uses Microsoft Windows.
- ___ (d) The application meets (or has an approved migration plan to meet) all of the following requirements:

- (1) Uses pointer shapes in table 2-1 of the DII Style Guide.
- (2) Uses the keyboard mappings in appendices A and B of the DII Style Guide.
- (3) Implements navigation, selection, activation, and interactive control specifications in sections 3.2, 3.3, 3.4, and 3.6 of the DII Style Guide.
- (4) Implements window specifications in sections 3.1 and 4.0 of the DII Style Guide.
- (5) Implements specifications for menus, controls, and vocabulary in sections 5.0 and 6.0 and appendix C of the DII Style Guide.

Level 5 Compliance

The segment is fully compliant with the style of the native GUI.

- ___ (a) The segment is fully compliant with level 1 style requirements.
- ___ (b) If X-based, the segment uses Motif version 1.2 or greater.
- ___ (c) If Windows-based, the segment uses the version provided with the COE.
- ___ (d) The segment meets (or has an approved migration plan to meet) all of the following requirements:
 - (1) Implements application window design specifications in section 8.0 of the DII Style Guide.
 - (2) Implements the default color set and resource settings in appendix D of the DII Style Guide.

The segment is launched from the same desktop provided with the COE.

- ___ (a) The segment is launched from a desktop configured in accordance with section 7.1 of the DII Style Guide.

Level 6 Compliance

The segment is either completely compliant with the DII Style Guide or has minimal deviations that have been approved by the DISA Chief Engineer.

- ___ (a) The segment is fully compliant with level 5 style requirements.
- ___ (b) If X-based, the segment uses the version of Motif provided with the COE.
- ___ (c) The segment can change color and font size dynamically using the desktop provided with the COE.
- ___ (d) The segment meets (or has an approved migration plan to meet) all of the following requirements:
 - (1) Supports integrated pointing device functions in table 2-1 of the DII Style Guide.
 - (2) Implements transfer specifications in section 3.5 of the DII Style Guide.
 - (3) Implements information presentation specifications in section 9.0 of the DII Style Guide.
 - (4) Implements task-specific window design specifications in section 10.0 of the DII Style Guide.
 - (5) Implements user support specifications in section 11.0 of the DII Style Guide.

Level 8 Compliance

The segment is fully compliant with the DII Style Guide.

- ___ (a) The segment is fully compliant with level 6 style requirements.

- ____ (b) The segment implements the application design specifications in section 7.2 of the DII Style Guide.

APPENDIX I

USER INTERFACE SPECIFICATIONS CHECKLIST

PURPOSE

The User Interface Specifications (UIS) checklist is provided as a tool to use in assessing the extent to which software complies with the specifications contained in the DII style guide. A completed checklist shall accompany the submission of segments to DISA and serve as documentation that the detailed requirements associated with the style-related compliance items in appendix H have been satisfied. The checklist can also be incorporated into the software quality assurance process of an organization and applied as needed during software development and testing to ensure that the user interface complies with DII style specifications. In this case, the use of the checklist is for the benefit of the organization performing the assessment to provide guidance for software development and improvement. While a partial assessment may be of value to the organization, using the checklist in this manner cannot be represented as an indication of DII compliance.

The checklist can be used to assess the degree of style compliance of a single segment or application or an entire system. While the checklist defines the required appearance and behavior for a wide range of user interface features, it does not mandate that all of these features be implemented in order to be considered DII compliant. It is expected that an organization will follow the tailoring process described below and then apply the resulting checklist to ensure that the features present in the software appear and behave according to DII specifications. A style assessment is the same whether the software being examined is a segment, an application, or a system; the only difference is in the scope of the assessment. The decision process should be to determine whether a given style feature is required for software usability and then to assess if the feature is present and has been implemented in a DII-compliant manner.

STYLE ASSESSMENT PROCESS

Tailoring the Checklist. To tailor the checklist, first review each of the major sections and determine which ones apply to the software being reviewed and which ones can be excluded from the assessment. Next, review the remaining sections and identify the specific items that are not applicable to the software being assessed and can also be excluded from the assessment. Finally, document the reasons for the omissions to establish a record of the logic of the tailoring process.

The tailoring process should be based on criteria such as (a) the operational capabilities being delivered as they are defined in requirements or design documentation for the software, (b) the maturity of the software (e.g., is it an initial prototype or a production-level delivery), and (c) the development priorities of the organization responsible for the software being evaluated. While some sections of the checklist can be omitted from the assessment because the style features described are inappropriate to implement in the software, items cannot be excluded solely because they happen to be absent from the user interface.⁵⁶ In addition, if the checklist will be submitted to DISA as justification for a particular compliance level, sections of the checklist cannot be excluded if they are required to satisfy a style item at that level. For example, a checklist that documents level 6 compliance cannot exclude the section related to object transfer since the style requirements for that level call for a segment to implement this user interface feature.

⁵⁶ More specific criteria on identifying non-applicable items will be provided in a future version of this document.

The tailoring process should be documented in enough detail that an outside reviewer can examine a completed checklist and determine if the resulting assessment provides an adequate evaluation of the level of DII style compliance. The documentation shall be submitted to DISA along with the completed checklist as justification for a particular compliance level. Those user interface features for which DISA has approved a waiver to diverge from DII requirements should be identified in the documentation.

Using the Checklist. To use the checklist, compare the appearance and behavior of the software with that specified in the checklist item. Code the item as YES if the software implements the item in the manner described. All instances of the style feature described by the item must be implemented in the software in the manner described for the item to be considered DII-compliant (i.e., be coded as YES). Code the item as NO if the software diverges from the appearance or behavior required by the item. Annotate the item with a description of the divergence (e.g., the window, menu, and/or button where the failure was observed) and, if appropriate, an explanation for why the divergence occurred (e.g., operational requirement), and include these explanations with the completed checklist. Indicate if the organization has an approved migration plan that will correct the divergence or if a waiver from DISA is requested to allow the divergence.

It is assumed that the individual using the checklist is familiar with the tasks performed by the software being assessed and can access documentation with which to confirm expected user interaction with the software. Information about the software can be obtained from interactions with the developer and the intended user community and from documents such as a user's guide and concept of operations. The relevant section of the style guide should be consulted if additional detail about individual checklist items is needed. Checklist items have been written to describe the appearance and behavior of the user interface as it is presented to users; the intent is to assess compliance through interaction with the user interface, rather than by review of software code.

The compliance assessment should be performed on the target hardware platform(s) and under environmental conditions similar to the operational setting(s) in which the software will be used. An appropriate test environment is important so that the adequacy of style features such as color, font, and contrast can be assessed. In addition, in cases where software is drawn from multiple sources, the assessment should include the new components developed by the organization performing the assessment as well as existing components that have been assembled and/or customized within the software. Components contributed by different development groups need to be assessed to ensure that a comprehensive review is performed of the software. A description of the test conditions should be included with the completed checklist, especially if the software could not be exercised in an operationally appropriate environment, so that an outside reviewer can determine the adequacy of the assessment.

The amount of the user interface (i.e., number of windows) that needs to be examined to establish compliance depends on the characteristics of the software being reviewed. It is expected that an assessment will examine at least 75 percent of the windows in the software and include all unique windows. In addition, all non-unique windows need to be identified, and the assessment is expected to include at least one example of each of them. Finally, the assessment needs to exercise every operational function performed by the software, even if the function is only an option within the software or is expected to be phased out. The assessment also needs to exercise all utility functions and known failure modes available within the software.

USER INTERFACE SPECIFICATIONS CHECKLIST

2.0 INPUT DEVICES

2.1 POINTING DEVICE INPUT

2.1.1 The Pointer

- _____ 1 The pointing device is associated with a single pointer on the screen.
- _____ 2 The hotspot of the pointer indicates the precise location where operations occur.
- _____ 3 The location of the hotspot does not move as the pointer changes shape.
- _____ 4 The pointer moves anywhere on the screen.
- _____ 5 When users move the pointing device, the pointer moves in the corresponding direction.
- _____ 6 The pointer remains in place until moved by users; it is not moved arbitrarily by the application.

2.1.2 Pointer Shapes

- _____ 1 The pointer shapes in figure 2-1 are used when providing the functions indicated.
- _____ 2 The application redefines pointer shape only when the pointer is in an application window.
- _____ 3 New pointer shapes are not created for functions that already have a shape.
- _____ 4 Pointer shapes are not associated with functions they were not designed to represent.
- _____ 5 New pointer shapes are easy to see, with a hotspot that is obvious and easy to locate.
- _____ 6 New pointer shapes suggest their purpose and are not confused with other objects.

2.1.3 Pointing Device Buttons

- _____ 1 The application supports integrated Select and Transfer functions as shown in table 2-1.
- _____ 2 Motif: When an integrated model is implemented, the application does not perform any Transfer function with button 2 when that button is used as BAdjust.
- _____ 3 The application does not change pointing device characteristics defined at the system level.

2.2 KEYBOARD INPUT

2.2.1 Fixed Function Keys

- _____ 1 The application uses the keyboard mappings in appendix B when performing the functions in appendix A.
- _____ 2 If the application defines additional functions, they do not conflict with keys listed in appendix A.
- _____ 3 <Ctrl>, <Shift>, and <Alt> are used only to modify the function of other key(s).
- _____ 4 <Shift>+key(s) are used for actions that extend or are complementary to the actions assigned to the unmodified key(s).

- _____ 5 <Ctrl>+key(s) are used for infrequent actions or for actions that represent larger-scale versions of the actions assigned to the unmodified key(s).
- _____ 6 <Alt> is used only to provide access to mnemonics.
- _____ 7 New key bindings are visible in application windows (e.g., as mnemonics/accelerators).

2.2.2 Variable Function Keys

- _____ 1 Command names for variable function keys are displayed on the screen as soft keys.
- _____ 2 Soft key labels displayed on the screen are modified when the meaning of a key changes.
- _____ 3 No more than two functions are assigned per soft key.
- _____ 4 Users can easily return to the set of base-level functions (e.g., a “Home” key is available).
- _____ 5 Soft keys are implemented in a manner consistent with input focus policy.
- _____ 6 The actions mapped to soft keys do not conflict with those in appendix A.
- _____ 7 Soft keys are grayed out when the window containing the keys does not have focus.

2.2.3 Text Entry

2.2.3.1 Text Entry Modes

- _____ 1 <Insert> toggles between insert and replace modes if the field supports replace mode.
- _____ 2 The application provides access to both insert and replace modes for text entry.
- _____ 3 The application does not switch arbitrarily between modes as users move between fields.

2.2.3.2 Text Entry Actions

- _____ 1 <Space> (or <Shift><Space> in Motif) inserts a space.
- _____ 2 <Enter> (or <Return> in Motif) inserts a carriage return in multi-line text.
- _____ 3 <Backspace> deletes the character to the left of the text cursor.
- _____ 4 <Delete> deletes the character to the right of the text cursor.
- _____ 5 <Tab> inserts a tab or moves to the next tab stop in multi-line text.
- _____ 6 Double clicking on text selects and highlights the word at the location of the pointer.
- _____ 7 When users highlight text and begin typing, the text disappears, the text cursor appears, and the new text is displayed.
- _____ 8 Motif: Triple clicking on text selects the line of text, and quadruple clicking selects multiple lines (e.g., a paragraph).

3.0 USER-COMPUTER INTERACTION

3.1 INPUT FOCUS

3.1.1 Focus Models

- _____ 1 Only one window on the screen has input focus at any time; within that window, only one object has focus at a time.
- _____ 2 The application is able to support both explicit and implicit focus.

Note: Because the default policy is assumed to be explicit, the checklist items below define an appearance and behavior based on this focus policy.

3.1.2 Assigning Focus with the Pointing Device

- ___ 1 Users assign focus by moving the pointer into the window and clicking BSelect.
- ___ 2 The window with focus is denoted by a change in the color of the window frame.

3.1.3 Assigning Focus with the Keyboard

- ___ 1 <Ctrl><Esc> displays a window listing currently running applications.
- ___ 2 <Alt><Esc> (and <Alt><Shift><Esc> in Windows) moves focus between applications.
- ___ 3 <Alt><Tab> and <Alt><Shift><Tab> move focus forward and backward through the window families in the application.
- ___ 4 <Alt><F6> and <Alt><Shift><F6> move focus forward and backward through the windows within a family.
- ___ 5 Windows: <Ctrl><F6> and <Ctrl><Shift><F6> move focus forward and backward through document windows in the application.
- ___ 6 <F6> (and <Shift><F6> in Windows) move focus between panes in the active window.

3.2 NAVIGATION

3.2.1 Pointing Device Navigation

- ___ 1 Placing the pointer on an object and clicking BSelect moves focus to the object; the object is also selected or activated (and highlights).
- ___ 2 The highlighting remains visible only while the window in which the object appears has focus.
- ___ 3 Clicking BSelect does not move focus to an object that is not selectable and cannot accept input.
- ___ 4 Placing the pointer on an object, pressing <Ctrl>, and clicking BSelect moves focus to the object but does not activate it.
- ___ 5 In scrollable objects, pressing BSelect and dragging the pointer outside the object executes autoscrolling (i.e., scrolls the object in the direction of the pointer).

3.2.2 Keyboard Navigation

3.2.2.1 The Location Cursor

- ___ 1 A single location cursor indicates the object with focus in a window.
- ___ 2 Only one location cursor appears in a window at any time.
- ___ 3 The position of the location cursor is not affected by movement of the pointing device.
- ___ 4 The shape of the text cursor is a vertical bar in insert mode and a shaded rectangle in replace mode.
- ___ 5 The text cursor flashes when the object containing it has keyboard focus.

- _____ 6 The flash rate for the text cursor is 2-5 Hz.
- _____ 7 The text cursor is easy to see whenever it appears in a text area.
- _____ 8 New location cursor shapes are created only if existing ones do not provide the functions desired.

3.2.2.2 Navigation in Tab Groups

- _____ 1 <Tab> (and <Ctrl><Tab> in Motif) moves the location cursor to the next tab group.
- _____ 2 <Shift><Tab> (and <Ctrl><Shift><Tab> in Motif) moves the location cursor to the previous tab group.
- _____ 3 If none of the controls in a tab group can have keyboard focus, <Tab> (and <Ctrl><Tab> in Motif) skips the group.
- _____ 4 The arrow keys move the location cursor among the elements of the tab group with focus.
- _____ 5 <Ctrl> in combination with the arrow keys move the location cursor one large increment (e.g., to the next word in text).
- _____ 6 If a graphics-like object uses a positional cursor, the arrow keys move the cursor one unit at a time in the direction indicated by the arrow.
- _____ 7 <Home> and <End> move the location cursor to the leftmost/rightmost element.
- _____ 8 <Ctrl><Home> and <Ctrl><End> move the location cursor to the beginning/end element.
- _____ 9 In scrollable objects, <PageUp>, <PageDown>, <Ctrl><PageUp>, and <Ctrl><PageDown> scroll one page (minus one line).
- _____ 10 Windows: When <ScrollLock> is enabled, the navigation keys scroll data without moving the location cursor or affecting existing selections.

3.2.2.3 Location Cursor Behavior During Navigation

- _____ 1 Tab group order is left to right, top to bottom in the window.
- _____ 2 When a window opens, the location cursor is on the control users are expected to interact with first.
- _____ 3 When a window regains focus, the location cursor is on the control that last had focus.
- _____ 4 Location cursor movement in a window is from upper left to lower right unless the control is scrollable.
- _____ 5 The location cursor wraps between the last and first control/groups in the window.
- _____ 6 When the location cursor moves to a tab group, it is placed on the first available control in the group.
- _____ 7 The location cursor skips a tab group if none of the controls can have keyboard focus.
- _____ 8 In scrollable controls, focus remains on the element where it was before scrolling began even though the location cursor may not be in view.
- _____ 9 In scrollable controls, any keyboard action that moves the location cursor or makes a modification at the cursor location scrolls the control so the cursor is visible.
- _____ 10 The location cursor is always visible as it moves between/within tab groups in a window.
- _____ 11 Moving the location cursor to a control does not change the size or position of the control.

- ___12 Motif: Keyboard navigation does not change select state; users execute a select action while location cursor is on a control to change its state.
- ___13 Windows: Keyboard navigation changes select state, unless scroll lock mode is enabled.

3.2.2.4 Text Cursor Behavior During Navigation

- ___1 When users click BSelect in an empty text area, the text cursor appears at the leftmost position in the area.
- ___2 When users click BSelect in a text area that contains text and the pointer is on the text, the text cursor is placed between the characters under the pointer.
- ___3 When users click BSelect in a text area that contains text and the pointer is beyond the end of text, the text cursor appears to the right of the final text character.
- ___4 <Tab> moves focus to a text field from the keyboard and displays the text cursor in the field.
- ___5 Motif: When users tab between text fields, the text cursor appears at the beginning of the text in the field.
- ___6 Windows: When users tab between text fields, the text cursor appears at the end of the text in the field.
- ___7 The arrow keys moves the text cursor one character or one line in the direction indicated by the arrow.
- ___8 The text cursor disappears from view when a text area loses focus and reappears at the same place when the area regains focus.
- ___9 If the cursor is not removed from view, it is grayed out and stops flashing when the area loses focus and returns to normal appearance and resumes flashing when the area regains focus.
- ___10 The text cursor appears only in text entry areas and not where text entry is not possible.
- ___11 Text entry is possible only after the text cursor is visible at a legal location; text entry is not possible when the text cursor is not visible.

3.3 SELECTION

3.3.1 Pointing Device Selection Methods

- ___1 The methods in table 3-1 are used for making selections in one- and two-dimensional collections.
- ___2 Motif: Users can perform the Adjust function in making a selection using either BSelect or BAdjust.
- ___3 Windows: Drag handles are displayed on or around an object when it is selected.
- ___4 The selection method(s) available in a collection match the type of selection that users are expected to make in the collection.
- ___5 Deselection affects only the select state of elements and does not delete any of the elements in a collection.
- ___6 Windows: <F8> toggles extend mode; when enabled, selection behaves as if <Shift> is locked down for all direction keys and pointing device actions.
- ___7 Windows: If selection modes are available, they supplement standard selection methods and include a visual cue when the mode is enabled.

3.3.2 Keyboard Selection Methods

- _____ 1 The location cursor is a solid rectangle in normal mode, a dotted rectangle in add mode.
- _____ 2 <Shift><F8> toggles between add mode and normal mode.
- _____ 3 Toggling between modes does not affect the select state of selections made with the pointing device.
- _____ 4 Motif: The methods in table 3-2 are used to make selections in one- and two-dimensional collections.
- _____ 5 Motif: The key bindings in appendix A are used to perform range selection in text collections.
- _____ 6 Windows: The methods in table 3-3 are used to make selections in one- and two-dimensional collections.
- _____ 7 <Backspace> and <Delete> delete the current selection.
- _____ 8 Motif: <Ctrl></> selects all of the objects in a collection.
- _____ 9 Motif: <Ctrl><\> deselects all of the objects in a collection.
- _____ 10 Motif: <Alt><Insert> reselects all elements in the most recently performed selection.

3.4 ACTIVATION

3.4.1 Basic Activation

- _____ 1 Placing the pointer on a button and clicking BSelect activates it.
- _____ 2 Pressing <Space> (and <Select> in Motif) on the button with focus activates it.
- _____ 3 When BSelect is pressed on a button, the appearance of the button changes (e.g., highlights) to indicate pending activation.
- _____ 4 When BSelect is pressed and the pointer is moved off a button, it returns to its normal appearance.

3.4.2 Mnemonics and Accelerators

- _____ 1 In the window with focus, <Alt> and the mnemonic for an object moves the location cursor to the object and selects or activates it.
- _____ 2 If the location cursor is already on an object, pressing the mnemonic for the object selects or activates it.
- _____ 3 If the menu or control is unavailable, pressing the mnemonic has no effect and focus remains unchanged.
- _____ 4 An accelerator is executed only when the window containing the accelerator has focus.

3.4.3 Default Actions

- _____ 1 When a default action is assigned to an object, the action is executed by double clicking BSelect on the object.
- _____ 2 If the object assigned a default action is used for making selections, the element under the pointer is selected and then the default action is executed.

- _____ 3 <Enter> (and <Ctrl><Return> in Motif) invokes the default action after making a selection in a window; these keys also invoke the default action in a window if focus is on an object other than multi-line text.
- _____ 4 If a default action is available, users can reverse the effects of the action (e.g., by selecting an Undo menu option or a Cancel push button).

3.4.4 Expert Activation, Previewing, and Autorepeat (Motif Only)

- _____ 1 If expert activation is implemented for an object, double clicking BSelect on the object performs the expert action.
- _____ 2 An expert action is available only in a group of push buttons or a group of radio buttons where one of the buttons is always on.
- _____ 3 If an expert action is available, it is used only as short-cut to features available elsewhere in a window.
- _____ 4 When focus is on a button used for expert activation, there is no default action available, unless the default and expert actions are the same.
- _____ 5 If previewing is available, pressing BSelect on a push button or toggle button displays information describing what happens when the button is activated; the information is removed when BSelect is released.
- _____ 6 If previewing is available, the application provides a means for expert users to disable it.
- _____ 7 If autorepeat is available, holding down BSelect on a push button executes its action repeatedly; autorepeat stops when BSelect is released.

3.4.5 Cancel Activation

- _____ 1 <Esc> (or <Cancel> in Motif) cancels the action being executed and returns the object to its state prior to the action.

3.5 TRANSFER

3.5.1 Drag Transfer

3.5.1.1 Drag Transfer Operations

- _____ 1 The application supports drag transfer operations.
- _____ 2 Motif: Drag transfer is available for all objects that are represented as icons and for all elements that users can manipulate.
- _____ 3 Motif: Drag transfer provides redundant access to functionality available elsewhere in the window (e.g., through menus or push buttons).
- _____ 4 Motif: To perform a drag move, users press <Shift> and drag the object using BSelect (or BTransfer).
- _____ 5 Motif: To perform a drag copy, users press <Ctrl> and drag the object using BSelect (or BTransfer).
- _____ 6 Motif: To perform a drag link, users press <Ctrl><Shift> and drag the object using BSelect (or BTransfer).

- ___ 7 Motif: Releasing BSelect (or BTransfer) ends a drag operation; the insertion point is the position where this action occurs.
- ___ 8 Motif: The default drag operation if no modifier key is used is a move.
- ___ 9 Motif: BSelect initiates a drag if the drag is started on a selected element; the drag starts once the drag threshold has been reached.
- ___ 10 Windows: To perform a drag transfer, users drag the object using BSelect.
- ___ 11 Windows: The default drag operation is a move but may be a copy or link as defined by the destination.
- ___ 12 <Esc> (or <Cancel> in Motif) cancels a drag operation that is in progress and returns the object being dragged to its original location; releasing BSelect (or BTransfer in Motif) when not over a drop site also ends a drag operation.
- ___ 13 Dragging a set of selected elements drags the entire selection.
- ___ 14 The elements moved within a collection remain selected after they have been moved.
- ___ 15 Motif: Dragging an unselected element drags only the element and does not affect the selection.
- ___ 16 Motif: Dragging in overlapping elements occurs on the highest draggable element in the stack.

3.5.1.2 Feedback During Drag Transfer

- ___ 1 During a drag operation, the pointer changes to a drag icon.
- ___ 2 Motif: The source indicator in the drag icon represents the type of object(s) being dragged.
- ___ 3 Motif: The operation indicator in the drag icon shows the type of drag operation being performed.
- ___ 4 Motif: The operation indicator is included in the drag icon if the operation is a copy or link but not if it is a move.
- ___ 5 Motif: The state indicator has an arrow shape when the icon is on a valid drop site and a “cannot” shape when it is on an invalid drop site.
- ___ 6 Windows: As the pointer moves, the object, its outline, or a representation of the object moves with the pointer.
- ___ 7 The drop site changes appearance when the drag icon is placed on it; options include showing a solid line around the site, a raised or lowered edge around the site, or a pixmap drawn over the site in Motif or highlighting or other form of visual emphasis in Windows.
- ___ 8 Motif: When the drag icon is placed on a drop site, the site has a beveled edge that makes the site look recessed.
- ___ 9 If a drag transfer is successful, the object is placed at the drop site and the drag icon is removed.
- ___ 10 If the transfer does not occur immediately, the pointer changes to a “busy” shape until the transfer is complete.
- ___ 11 If a drag transfer fails, the object remains at the drag source and the drag icon is removed.
- ___ 12 Motif: The drag icon “melts into” the drop zone when dropped on a valid site and “snaps back” to the drag site when dropped on an invalid site.

- ____ 13 Motif: If a drag transfer fails, an error message is displayed that informs users why the failure occurred.
- ____ 14 Motif: If multiple elements are involved in a drag operation and the transfer is only partially successful, this feedback indicates which transfers succeeded and which failed.
- ____ 15 Motif: If the application provides help on a drag transfer, the dialog window providing this information contains a Cancel push button for cancelling the transfer operation.

3.5.1.3 Performance Guidelines (Motif Only)

- ____ 1 When BSelect is pressed, a drag transfer is initiated when the pointer has moved 10 pixels; when BTransfer is pressed, there is no drag threshold. A drag icon is displayed within 50 msec (max 70 msec) after the drag is initiated.
- ____ 2 When the drag icon is moved over a drop zone, the drag icon or drop zone change appearance within 50 msec (max 70 msec).
- ____ 3 When the drag icon is dropped on a drop zone, feedback occurs within 50 msec (max 120 msec) of releasing the pointing device button. The visual effects last 200-350 msec (max 500 msec).
- ____ 4 Data transfer is completed in 0.3-1.0 sec (max 2 sec) after the drop occurred. If the transfer takes longer than 2 sec, the pointer changes to a “busy” shape whenever it is on the object where the transfer is occurring, and a working message window is displayed and its contents updated every 2-3 seconds until the transfer is completed.

3.5.2 Clipboard Transfer

- ____ 1 The application supports clipboard transfer operations.
- ____ 2 If access to clipboard transfer is provided in an Edit menu, the Cut, Copy, and Paste options execute these actions as defined in appendix C.
- ____ 3 The keyboard accelerators in appendix C are available for executing cut, copy, and paste actions from the keyboard whenever an object that can be edited has keyboard focus.
- ____ 4 If the clipboard contents are text, Paste copies the contents to the location of the text cursor, and any existing text appears to the left of the cursor. If text has been previously selected when a Paste occurs, the selected text is deleted and the clipboard contents are pasted at that location.
- ____ 5 If the clipboard contents are graphic, Paste copies the contents to the pointer location in the window with input focus.
- ____ 6 If Paste is invoked from the pop-up menu for an object, the clipboard contents are pasted at the insertion point of the object. If the menu is popped up over a selection, the selection is deleted and the clipboard contents replace it if possible.
- ____ 7 Pasting an object from the clipboard does not select the object.
- ____ 8 The pasted object remains in the clipboard until another object is cut or copied into it.
- ____ 9 Users can view clipboard contents and are informed when they cut or copy an object whose size exceeds clipboard capacity.

3.5.3 Primary Transfer (Motif Only)

- _____ 1 To perform a primary move, users select the object, place the pointer at the destination, hold down <Shift>, and click BTransfer.
- _____ 2 To perform a primary copy, users select the object, place the pointer at the destination, hold down <Ctrl>, and click BTransfer.
- _____ 3 To perform a primary link, users select the object, place the pointer at the destination, hold down <Ctrl><Shift>, and click BTransfer.
- _____ 4 The insertion point for primary transfer operations is the position where BTransfer is released.
- _____ 5 If access to primary transfer is provided in an Edit menu, the Primary Copy and Primary Move options execute the actions and use the keyboard accelerators defined in appendix C.
- _____ 6 Transferring an object by primary copy or primary link does not select the object.
- _____ 7 Transferring an object by a primary move selects the object.
- _____ 8 The default if no modifier key is used with BTransfer is a copy operation.

3.5.4 Quick Transfer (Motif Only)

- _____ 1 Quick transfer is available in text components in the application.
- _____ 2 To perform a quick move, users hold down <Alt><Shift> while dragging the object using BTransfer.
- _____ 3 To perform a quick copy, users hold down <Alt><Ctrl> while dragging the object using BTransfer.
- _____ 4 To perform a quick link, users hold down <Alt><Ctrl><Shift> while dragging the object using BTransfer.
- _____ 5 The default when <Alt> and BTransfer are used to perform a quick transfer is a Copy operation.
- _____ 6 Using quick transfer does not select the object being transferred.

3.6 INTERACTIVE CONTROL

3.6.1 Object-Action Selection Model

- _____ 1 When working in the application, users first select an object, then select an action to perform on that object.
- _____ 2 When users make or change a selection, no other action is performed on the selected elements (i.e., separate selection and activation actions).
- _____ 3 The application informs users when interaction diverges from the object-action selection paradigm.

3.6.2 User Control of Interaction

- _____ 1 The application executes an action only in response to explicit user input.
- _____ 2 The pace of user input does not slow down the speed of application processing.

3.6.3 Immediate Feedback

- _____ 1 Some visible response by the application occurs within 0.2 sec of any user action.
- _____ 2 When a user request takes longer than 2-3 sec but less than 10 sec to process, the pointer changes to a “busy” shape.
- _____ 3 When a user request takes more than 10 sec to process, a working message window is displayed.
- _____ 4 Error feedback is provided to users within 2 sec of the time the error was detected.
- _____ 5 Visual cues show when the application can accept input, when it is temporarily unavailable, or it is unavailable during extended processing.
- _____ 6 The appearance of objects provides an indication of their availability.
- _____ 7 If an operation requires several actions, users are prompted with the actions to take.
- _____ 8 The application ignores user actions made during periods when input cannot be accepted.
- _____ 9 The pointing device and/or keyboard are disabled when input may be destructive; users cannot override disabling but are able to stop a process.

3.6.4 Error Detection

- _____ 1 The application does not execute an invalid action except to display an error message.
- _____ 2 When users make multiple errors with a single action, they are notified of each error.
- _____ 3 Feedback is immediate (i.e., occurs within 2 sec of error detection), is visual and/or auditory, and explains the nature of the error.
- _____ 4 When an error is repeated, feedback shows that the attempted correction was processed.
- _____ 5 Users are required to correct only the invalid action and not to repeat the entire sequence.
- _____ 6 After making correction, users execute same action for re-entry that was used originally.

3.6.5 Explicit Destruction

- _____ 1 Users confirm a destructive action before the action is executed by the application.
- _____ 2 Users confirm a close-window action only when the action will cause loss of data.
- _____ 3 The window with the destructive action remains displayed while request to confirm the action is presented.
- _____ 4 If the destructive action applies to multiple objects, users are provided with a list of the objects from which to select the ones that apply.

3.6.6 General “Undo” Capability

- _____ 1 Users can undo the most recent selection or action unless it was one requiring explicit destruction.
- _____ 2 Irreversible actions are labeled and clearly separated from those that are not irreversible.

3.6.7 Use of Processing Modes

- _____ 2 If a processing mode is implemented, the application provides a visual cue to indicate the mode currently in effect.

3.6.8 Consistency in Performing Operations

- _____ 1 The sequence of actions executed by users to perform the operations in the application can be completed rapidly and efficiently (e.g., with minimum keystrokes).
- _____ 2 Users execute the same sequence of actions whenever they perform an operation in the application.
- _____ 3 Users execute only those actions that are required to perform an operation.

4.0 WINDOWS AND WINDOW ICONS

4.1 WINDOW MANAGEMENT

4.1.1 Window Components

4.1.1.1 Title Bar

- _____ 1 The title bar extends across the top of the window, with the window name in the middle of the bar.
- _____ 2 Clicking on the title bar raises the window to the front of the screen and gives it focus.
- _____ 3 Pressing BSelect or BTransfer on the title bar and then dragging the pointer moves the window as the pointer moves.

4.1.1.2 Window Menu

- _____ 1 If a window supports any window management functions, it has a Window menu with options for performing these functions.
- _____ 2 The Window menu button is located at the left edge of the title bar.
- _____ 3 If a window management function is included in the Window menu, it executes the action and includes the mnemonic in appendix C.
- _____ 4 An accelerator is included only for the Close option in the Window menu.
- _____ 5 If the application includes accelerators for Window menu options, it uses the ones listed in appendix C.
- _____ 6 Motif: If any of the following window management functions is included in the Window menu, they are ordered: Restore, Move, Size, Minimize, Maximize, Lower, Occupy Workspace, Occupy All Workspaces, Unoccupy Workspace, and Close.
- _____ 7 Motif: Separators are included after the Lower option and before the Close option.
- _____ 8 Windows: If any of the following window management functions is included in a Window menu, they are ordered: Restore, Move, Size, Minimize, Maximize, Close, Switch To, and Next.
- _____ 9 Windows: Separators are included before the Close option and after it if additional options are included in the menu.
- _____ 10 If the application adds functions to the Window menu, they are appended to the bottom of the menu, with a separator between Close and the added options.
- _____ 11 Spring-loaded and posted methods are used to display Window menu and activate options with the pointing device.
- _____ 12 Double clicking on the Window menu button closes the window.

- ___ 13 <Alt><Space> (or <Shift><Escape> in Motif) displays the Window menu in the application window with input focus.
- ___ 14 Windows: <Alt><Hyphen> displays the Window menu in the document window with input focus.
- ___ 15 The arrow keys move the location cursor between available options in the Window menu.
- ___ 16 <Enter> (or <Return>, <Select>, or <Space> in Motif) activates an option and dismisses the Window menu.
- ___ 17 <Esc> (or <Cancel> in Motif) dismisses the Window menu without activating an option.

4.1.1.3 Window Control Buttons

- ___ 1 The Maximize button is positioned at the right edge of the title bar.
- ___ 2 The graphic in the Maximize button is a large square in Motif or an up arrow in Windows.
- ___ 3 Motif: Activating the Maximize button in a normal-size window expands it to its maximum size.
- ___ 4 Motif: Activating the Maximize button in a maximized window restores the window to its size and location before being maximized.
- ___ 5 Windows: Activating the Maximize button in a normal-size window expands it to its maximum size and replaces the button with a Restore button.
- ___ 6 Windows: Activating the Restore button of a maximized window restores the window to its size and location before being maximized.
- ___ 7 The Minimize button is positioned to the left of the Maximize button in the title bar.
- ___ 8 The graphic in the Minimize button is a small square in Motif or a down arrow in Windows.
- ___ 9 Activating the Minimize button minimizes the window.
- ___ 10 If additional window functions are needed, they are mapped to buttons placed to the left of the Minimize button or to the right of the Window menu button.

4.1.1.6 Resize Border

- ___ 1 Window components are not placed outside the boundary formed by the resize borders.
- ___ 2 Pressing BSelect or BTransfer on the resize border and dragging the pointer changes the window size as the pointer moves.
- ___ 3 Windows: All windows have frames except when they are maximized and fill the entire screen.

4.1.2 Behavior in Window Families

4.1.2.1 Parent-Child Relationships

- ___ 1 Minimizing a primary window replaces it and all of its children with a window icon; processing in the window continues.
- ___ 2 Opening a window icon redisplay the primary window and its children.
- ___ 3 Each primary window in the application is minimized separately.

- ___ 4 When a primary window is raised or lowered, it and all of its children move with it.
- ___ 5 Closing a primary window removes it and its children from the screen; processing in the window stops.
- ___ 6 When the last primary window for an application is closed, the application is closed.
- ___ 7 Windows: A document window is a parent but also the child of the application window within which it is displayed.
- ___ 8 When secondary or dialog window is opened, it appears in front of the parent which stays displayed.
- ___ 9 When a secondary window is closed, its children are closed but its parent is not affected.
- ___ 10 A dialog window has no child windows (except a help window, if one is available).

4.1.2.2 Interaction Modes

- ___ 1 Motif: Child windows are modal only when the application cannot proceed without additional information (e.g., potential loss of data).
- ___ 2 Windows: Child windows are modal.

4.1.3 Window Management Considerations

4.1.3.1 Window Size

- ___ 1 The normal size of a window is large enough so that all objects in it are visible when the window first appears.
- ___ 2 The minimum size of a window is wide enough to read the title and tall enough to read the title and menu bar.
- ___ 3 When a window is maximized, the size of objects in the window does not change.
- ___ 4 When a window is resized, the size of objects in the window and their relative position do not change.
- ___ 5 The contents of the window remain visible during resizing so users can view the effect.
- ___ 6 A window can be resized larger only if more information can be displayed.
- ___ 7 Resizing larger and maximizing do not obscure restricted portions of the screen.

4.1.3.2 Window Arrangement

- ___ 1 Window placement is overlapping.

4.1.3.3 Window Positioning

- ___ 1 When initially displayed, a window is positioned on the screen so that it is completely visible.
- ___ 2 If a window is related to other windows already displayed, it is offset below and to the right (or to the left/below/above if space is limited) of the information to which it relates and does not obscure the title of the window with the information.
- ___ 3 If a new window is unrelated to other windows currently open, it is placed in the center of the screen (or in the center of the application window in Windows).

- _____ 4 If a dialog window (or a menu window in Motif) is already displayed but obscured by other windows, re-choosing the command that displayed the window raises it to the front of the window stack without affecting its position on the screen.

4.1.3.4 Processing in Minimized Windows

- _____ 1 Motif: A message window is displayed when critical events occur in a minimized window.
- _____ 2 Windows: The window icon flashes when critical events occur in a minimized window, with a message window displayed when the user restores the window.

4.1.3.5 Moving Windows Between Workspaces (Motif Only)

- _____ 1 When the application opens a new window, it is displayed in the user's current workspace and only occupies that workspace.
- _____ 2 When users move application windows between workspaces, the windows related to a particular task move together.

4.2 WINDOW ICONS

4.2.1 Appearance

- _____ 1 A window icon has the same title as its corresponding window.
- _____ 2 Motif: The graphic image for a window icon is the same as that used for the application icon on the desktop.
- _____ 3 Motif: When an icon does not have focus, its title is the same width as the icon image; the title may be truncated to fit.
- _____ 4 Motif: When an icon has focus, the location cursor is displayed on the icon and the full icon title is shown.
- _____ 5 Motif: The Window menu for a window icon has the same options (except Size) as the menu for the corresponding window.
- _____ 6 Motif: If Minimize is included in the Window menu for a window icon, it is not available for selection.

4.2.2 Behavior

- _____ 1 Double clicking on a window icon restores a minimized window and its children.
- _____ 2 If the window had been maximized prior to being minimized, double clicking on the icon displays the window in its maximized size.
- _____ 3 Placing the pointer on an icon and dragging it with BSelect or BTransfer moves the icon.
- _____ 4 Motif: Clicking BSelect on the icon displays the Window menu.
- _____ 5 Motif: Clicking BSelect anywhere outside the Window menu dismisses the menu.
- _____ 6 Motif: Navigating to an icon from the keyboard gives the icon focus and displays the Window menu.

- _____ 7 Motif: Selecting Maximize from the Window menu displays the window in its maximized state.

5.0 MENUS

5.1 PULL-DOWN MENUS

5.1.1 Appearance

- _____ 1 The menu title is displayed in a menu bar at the top of the window.
- _____ 2 The menu title describes the category or type of options presented in the menu.
- _____ 3 The menu title is different from the other titles in the menu bar and does not appear as an option in the menu itself.
- _____ 4 The menu title is a single word whenever possible and does not contain numbers.
- _____ 5 The first letter of each word in the menu title is capitalized, except for prepositions and articles.
- _____ 6 If the menu title contains an acronym, it is capitalized.
- _____ 7 The menu title does not contain an ellipsis or a right-pointing arrow.
- _____ 8 A routing option that displays a window is followed by an ellipsis.
- _____ 9 A routing option that displays a submenu is followed by a right-pointing arrow.
- _____ 10 A submenu appears to the right of the parent menu (or below if space to the right is limited).
- _____ 11 A submenu is positioned to align its first option with the arrow in the parent option.
- _____ 12 The parent option for a submenu is always displayed as available.
- _____ 13 When the parent option is activated, the submenu is displayed, even if all of the options are unavailable.
- _____ 14 A submenu contains only the options in the submenu; it does not repeat the parent option.

5.1.2 Behavior

- _____ 1 Spring-loaded and posted methods are used to display a menu and activate an option.
- _____ 2 <F10> (or <Shift><Menu>) moves the location cursor to the first available menu title in a menu bar; if none of the menu titles is available, these keys do not move the location cursor.
- _____ 3 <Left> and <Right> move the location cursor between available menu titles, with wrapping between the last and first titles.
- _____ 4 <F10> (or <Shift><Menu>) moves the location cursor from the menu bar to the previous object with focus.
- _____ 5 <Down> displays the menu for the title containing the location cursor.
- _____ 6 The arrow keys move the location cursor between available options in the menu, with wrapping between the last and first options.
- _____ 7 <Right> displays a submenu if the option with the location cursor is the parent for the submenu.
- _____ 8 <Left> dismisses a submenu and returns the location cursor to its parent option.

- ___ 9 <Enter> (or <Return>, <Select>, or <Space> in Motif) activates an option and dismisses the menu.
- ___ 10 <Esc> (or <Cancel> in Motif) dismisses the menu without activating an option and returns the location cursor to the object that previously had focus.

5.2 POP-UP MENUS

5.2.1 Appearance

- ___ 1 A pop-up menu provides redundant access to frequently executed actions.
- ___ 2 Motif: A pop-up menu includes a title describing the function performed by the menu or the object to which the menu relates.
- ___ 3 Motif: The title is displayed at the top of the menu and separated from the first menu option by a separator line.
- ___ 4 If a pop-up menu contains options taken from different menus in the window, options pertaining to the object to which the menu relates are presented first, followed by edit options and then by application-specific options. A separator delimits each group of options.
- ___ 5 The pop-up menu does not contain any submenus.
- ___ 6 Motif: If a pop-up menu includes any of the following options, they are ordered: Properties, Undo, Primary Move, Primary Copy, Primary Link, Cut, Copy, Copy Link, Paste, Paste Link, Clear, Delete, Select All, Deselect All, Select Pasted, Reselect, Promote.
- ___ 7 Motif: If the menu relates to a selectable object, it also includes Move To, Copy To, Put in Workspace, Delete, and Help options.
- ___ 8 When the pointing device is used to display a pop-up menu, its contents relate to the object under the pointer.
- ___ 9 When the keyboard is used to display a pop-up menu, its contents relate to the object with the location cursor.
- ___ 10 A pop-up menu is displayed near the object with which it is associated.

5.2.2 Behavior

- ___ 1 Users display a pop-up menu by pressing or clicking BMenu.
- ___ 2 Users navigate within a pop-up menu and activate an option with either BSelect or BMenu.
- ___ 3 Releasing BMenu with the pointer in the area that activated the menu dismisses it without activating an option.
- ___ 4 Motif: Clicking BSelect or BMenu outside the menu also dismisses it.
- ___ 5 Motif: When the pop-up menu relates to a selection, the action executed by the option chosen applies to all of the elements in the selection.
- ___ 6 Motif: When a pop-up menu relates to an unselected object, the action executed applies only to that object and not to any other object(s) that might be selected.
- ___ 7 Windows: If there is no current selection, clicking BMenu on an object both selects it and displays its pop-up menu.

- ___ 8 Windows: Clicking BSelect outside the menu removes the menu and selects the object under the pointer if one is there; clicking BMenu outside the menu only removes the menu.
- ___ 9 Motif: <Shift><F10> or <Menu> displays a pop-up menu if one is available.
- ___ 10 Windows: Keyboard activation of pop-up menus is not available.
- ___ 11 The arrow keys move the location cursor between available options in a pop-up menu.
- ___ 12 <Enter> (or <Return>, <Select>, or <Space> in Motif) activates an option and dismisses a pop-up menu.
- ___ 13 <Esc> (or <Cancel>, <Menu>, or <Shift><F10> in Motif) dismisses a pop-up menu without activating an option, and the location cursor returns to the object that had focus before the menu was displayed.

5.3 TEAR-OFF MENUS (Motif Only)

5.3.1 Appearance

- ___ 1 A tear-off menu is used when users need to select repeatedly from a pull-down menu, submenu, or pop-up menu.
- ___ 2 A tear-off menu contains a tear-off button with a dashed-line graphic that is the first menu option below the title.
- ___ 3 The tear-off button behaves in the same manner as other available options in the menu.
- ___ 4 When a tear-off menu is displayed, the location cursor is on the second available option if the first option is the tear-off button.
- ___ 5 The title of the menu window is the title of the associated tear-off menu.
- ___ 6 The contents of the menu window are the same as the original menu and in the same order but the tear-off button is not displayed in the window.
- ___ 7 The options in a menu window are dimmed when unavailable.
- ___ 8 The options in a menu window behave in the same manner as options in the menu (e.g., wording changes when activated).

5.3.2 Behavior

- ___ 1 Clicking BSelect on the tear-off button dismisses the menu and displays a menu window at the menu location.
- ___ 2 Dragging the tear-off button with BTransfer displays a menu window and moves it to a new location.
- ___ 3 When a menu window is displayed, it has focus, with the location cursor on the first available menu option.
- ___ 4 Clicking BSelect on an option in a menu window moves the location cursor to the option and executes it, and the window remains displayed.
- ___ 5 The arrow keys move the location cursor to the tear-off button in a menu.
- ___ 6 When focus is on the tear-off button, <Return>, <Enter>, <Select>, or <Space> dismisses the menu and displays a menu window at the menu location. The window receives focus, with the location cursor on the first available menu option.
- ___ 7 The arrow keys move the location cursor between available options in the menu window; <Return>, <Enter>, or <Space> activates an option.

- ___ 8 A menu window can be moved but not minimized or maximized.
- ___ 9 A tear-off menu can be displayed while the menu window is open, and options can be selected from either the menu or the window.
- ___ 10 When the menu is torn off a second time, the original menu window is dismissed and a new window appears.
- ___ 11 Selecting Close in the Window menu or pressing <Esc> (or <Cancel> in Motif) when the window has focus dismisses a menu window.

5.4 OPTION MENUS (Motif Only)

5.4.1 Appearance

- ___ 1 The title of an option menu is placed to the left of the option button (above the button if space is limited) and followed by a colon.
- ___ 2 The label in the option button is the most recent option selected from the menu and includes a bar graphic.
- ___ 3 The option button is large enough to display both the longest menu option and the bar graphic; the graphic does not obscure any text displayed in the button.

5.4.2 Behavior

- ___ 1 Spring-loaded and posted methods are used to display an option menu and select an option using BSelect or BMenu.
- ___ 2 When an option is selected in option menu, it appears as the label in the option button and the menu is dismissed.
- ___ 3 When focus is on an option menu, <Space> or <Select> displays an option menu, with the location cursor on the previously selected option in the menu.
- ___ 4 The arrow keys move the location cursor between available options in an option menu.
- ___ 5 <Return>, <Enter>, <Select>, or <Space> activates an option, dismisses the menu, and displays the option in the option button.
- ___ 6 <Esc> or <Cancel> dismisses an option menu without activating an option.

5.5 MENU DESIGN GUIDELINES

5.5.1 Format of Menu Options

- ___ 1 Menu options are presented in mixed case, with the first letter of each word capitalized, except for prepositions and articles.
- ___ 2 If the option contains an acronym, it is capitalized.
- ___ 3 The menu is wide enough to accommodate the widest option and an accelerator (if one is available).
- ___ 4 Accelerators include a plus sign to indicate the keys to be pressed at the same time.
- ___ 5 The end of the menu option is separated from the start of the accelerator by at least three character widths.
- ___ 6 Menu options and accelerators are left-justified and appear on a single line.

5.5.2 Wording of Menu Options

- ___ 1 Menu options are phrased to reflect the action executed and worded in user vocabulary.
- ___ 2 The vocabulary in appendix C is used when the actions it contains are used in menu options.
- ___ 3 The wording of an action toggle option reflects the action implemented when the option is activated.
- ___ 4 The wording of an action toggle option is semantically congruent with natural usage.
- ___ 5 When an action toggle option is activated, its wording changes to reflect the action that will be executed when the option is activated again.
- ___ 6 Only one of the actions for an action toggle option appears in the menu at any time.
- ___ 7 The wording of an Undo option changes dynamically to reflect the action to be undone.
- ___ 8 The wording of a state toggle option describes the state being set.
- ___ 9 Motif: A state toggle option includes a radio or check button to the left of the option label.
- ___ 10 Motif: When a state toggle option is activated, the select state of the radio or check button changes but the wording of the option remains the same.
- ___ 11 Motif: When a radio or check button appears in a menu, it always appears as either selected or unselected and is not removed from the menu when deselected.
- ___ 12 Windows: A state toggle option does not include a radio or check button to the left of the option label.
- ___ 13 Windows: When users activate an option in a set of nonexclusive settings, a check mark is placed to the left of the option; when users activate an option in a set of exclusive settings, a dot is placed to the left of the option.
- ___ 14 Windows: When the state of a setting is indeterminate, the graphic(s) is removed from all groups for which the settings are indeterminate.
- ___ 15 If radio buttons are used in a menu, they are grouped together and delimited from other menu options by a separator.
- ___ 16 The wording of each menu option is consistent in grammatical style and matches the corresponding menu title.
- ___ 17 Verbs are used as the first word in options in pull-down and pop-up menus.

5.5.3 Grouping Options into Menus and Submenus

- ___ 1 A pull-down or pop-up menu contains no less than two or more than 15 options.
- ___ 2 An option menu contains no more than 10-12 options.
- ___ 3 Menus with more than 4 options are divided into groups based on function, with each containing no more than 4 options (unless more are logical) and a separator line between groups.
- ___ 4 If menu options are not in functional groups, order is by frequency of usage, with the most frequently executed at the top and least frequently executed at the bottom.
- ___ 5 If menu options are not ordered in logical groups or by frequency, they are presented in alphabetical or numerical order.
- ___ 6 Destructive options are at the bottom of the menu.
- ___ 7 Options that perform opposing actions are not placed adjacent to each other.

- ___ 8 If similar options are in different menus, the options are ordered in a consistent manner in each menu.
- ___ 9 Submenus are limited to two levels below the parent menu option.
- ___ 10 Submenus contain at least three options.

5.5.4 Availability of Menu Options

- ___ 1 If an option is never available to a user, it is not included in a menu.
- ___ 2 If an option is temporarily unavailable, it is included in a menu but dimmed.

5.5.5 Mnemonics and Accelerators

- ___ 1 The application uses the mnemonics in appendix C whenever it implements any of the actions listed in the appendix.
- ___ 2 Mnemonics are available in all pull-down menu titles and options.
- ___ 3 Motif: Mnemonics are available in pop-up and tear-off menus.
- ___ 4 Windows: Mnemonics are not available in pop-up menus.
- ___ 5 The character assigned as the mnemonic is included in the text label and is underlined.
- ___ 6 Whenever possible, the mnemonic is the first letter of a menu title or option; a distinctive consonant in the label is preferred over vowels.
- ___ 7 If the mnemonic is not the first character, it is the last character of the label or the first character of the second word (if there is more than one word) or a sequential number that is assigned to the label.
- ___ 8 If the mnemonic does not appear in a menu title or option, it is placed in parentheses following the label.
- ___ 9 Mnemonics are not case-sensitive.
- ___ 10 The characters selected as mnemonics in each menu bar and each menu are unique.
- ___ 11 A menu title or option has the same mnemonic whenever it appears in the application.
- ___ 12 The same mnemonic is not used for options performing opposite or contradictory actions in different menus.
- ___ 13 Pull-down menus include accelerators for frequently executed menu options.
- ___ 14 The same key combination is used for an accelerator throughout the application.
- ___ 15 Motif: Accelerators are available in pop-up menus only if the menu contains the same options and accelerators as the corresponding pull-down menu.
- ___ 16 Windows: Accelerators are not available in pop-up menus.
- ___ 17 When an accelerator is activated, the associated menu is displayed briefly and its action is applied to the window that has focus.
- ___ 18 The application uses the accelerators in appendix C whenever it implements any of the actions listed.
- ___ 19 When new accelerators are created, they are of the form “modifier+character.”
- ___ 20 Similar key(s) are used in the mnemonic and accelerator for a menu option.

6.0 CONTROLS

6.1 PUSH BUTTONS

6.1.1 Appearance

- ___ 1 A push button is used to initiate an action.
- ___ 2 A push button contains a text or graphic label, centered in the button, indicating the action executed when the button is activated.
- ___ 3 The default push button has an extra border around it.
- ___ 4 All of the push buttons in a group are the same size and wide enough to display the longest text label or largest graphic.
- ___ 5 The first letter of each word in the text label is capitalized, except for prepositions and articles.
- ___ 6 The text label includes an ellipsis if activating the push button results in another window requesting additional information being displayed.
- ___ 7 The vocabulary in appendix C is used when the actions it contains are presented in push buttons.
- ___ 8 When new vocabulary is created, it describes actions not listed in appendix C.
- ___ 9 Push button labels are verbs, stated in active voice, and describe the action taken when the button is selected.
- ___ 10 The names of actions are congruent.
- ___ 11 The same vocabulary is used to describe an action throughout the application.
- ___ 12 When “All” is used in a label, there is no ambiguity as to the referent; labels with multiple referents include the name of the object/element.
- ___ 13 Action icons have unique graphics that are unambiguous and easily distinguished from other icon graphics.
- ___ 14 Action icons representing opposite actions have graphics that mirror each other.
- ___ 15 Action icons do not contain an ellipsis, even if activating the icon displays a window.

6.1.2 Behavior

- ___ 1 Placing the pointer on a push button and clicking BSelect activates a push button.
- ___ 2 When the location cursor is on a push button, <Space> (or <Ctrl><Space> in Motif) activates the button from the keyboard.
- ___ 3 When a push button is activated, it highlights and the action it represents is executed.
- ___ 4 Releasing BSelect outside the push button does not activate the push button.
- ___ 5 <Enter> (or <Return> in Motif) activates the default push button in a window.

6.2 RADIO BUTTONS

6.2.1 Appearance

- ___ 1 Radio buttons are used in groups to select one from a set of mutually exclusive options.
- ___ 2 The radio button label defines the state being set by the user.
- ___ 3 The first letter of each word in the label is capitalized, except for prepositions and articles.
- ___ 4 If a radio button cannot be selected, its label is grayed out to indicate its unavailability.
- ___ 5 Motif: When a group of radio buttons is displayed, one of the buttons in the group is selected.

- _____ 6 Motif: Users cannot deselect all of the radio buttons in a group; instead, a radio button labeled None is provided as one of the options in the group.
- _____ 7 Windows: In radio buttons that represent a fixed set of attributes for selection, the radio button corresponding to the current attribute is selected when initially displayed.
- _____ 8 Windows: In radio buttons that represent a heterogeneous set of attributes for selection, all of the radio buttons in the group are deselected when initially displayed, and choosing any button applies the option to the entire selection.

6.2.2 Behavior

- _____ 1 Placing the pointer on a radio button or its label and clicking BSelect selects the button.
- _____ 2 Motif: Moving the location cursor to a radio button and pressing <Space> or <Select> selects the button from the keyboard.
- _____ 3 Windows: Moving the location cursor to a radio button also selects it from the keyboard.
- _____ 4 When a radio button is selected, it highlights and any other selected button in the group is deselected.
- _____ 5 When a radio button is selected, only its select state changes; no action is taken, and no window is opened.
- _____ 6 If a radio button is in a window with a default action, pressing <Enter> (or <Return> in Motif) in a radio button selects the button and executes the action.

6.3 CHECK BUTTONS

6.3.1 Appearance

- _____ 1 Check buttons are used singly or in groups to indicate a nonexclusive setting; selecting one toggles a setting or state.
- _____ 2 The check button label defines the state being set by the user.
- _____ 3 The first letter of each word in the label is capitalized, except for prepositions and articles.
- _____ 4 If a check button cannot be selected, its label is grayed out to indicate its unavailability.
- _____ 5 A check button (not two radio buttons) is used if an option can only be set to on or off.
- _____ 6 Windows: When the state of the check button for a property is indeterminate, it is filled with a gray pattern.
- _____ 7 Windows: Clicking on a check button with an indeterminate state cycles through three states, with the button indicator reflecting the current state.

6.3.2 Behavior

- _____ 1 Placing the pointer on a check button or its label and clicking BSelect selects the button.
- _____ 2 When the location cursor is on a check button, <Space> (or <Select> in Motif) selects the button from the keyboard.
- _____ 3 When a check button is selected, it highlights and includes a check mark (in Motif) or is filled with an X (in Windows), and any other selected button remains selected.

- ___ 4 When a check button is selected, only its select state changes; no action is taken, and no window is opened.
- ___ 5 If a check button is in a window with a default action, pressing <Enter> (or <Return> in Motif) in a check button selects/deselects the button and executes the action.

6.4 LABELS

- ___ 1 A label is used to display static text and graphics in a window.
- ___ 2 Static text (e.g., titles, headings, and directions) is presented in a label and not in a text field.
- ___ 3 A label cannot be selected, and it is not traversable from the keyboard.
- ___ 4 The first letter of each word in the label is capitalized, except for prepositions and articles.

6.5 TEXT FIELDS

6.5.1 Appearance

- ___ 1 A text field is used to enter and edit text.
- ___ 2 A text field includes a label describing what is to be entered in the field.
- ___ 3 The label appears to the left or above the field and has the same background color as the window in which it appears.
- ___ 4 The label is followed by a colon, and the space between it and the text field is empty.
- ___ 5 The label is grayed out if the text field is unavailable for text entry.
- ___ 6 Cues regarding the format of the text to be entered are presented in the message bar of the window if one is present; otherwise, these cues are provided in the label.
- ___ 7 If a unit of measurement is always used, it is part of the label and does not have to be entered.
- ___ 8 Cues regarding whether text entry is mandatory or optional are presented in the message bar if the window includes one; otherwise, these cues are provided in the label or are color coded in the text field itself.
- ___ 9 The label is worded to be clearly different from the labels for other text fields in the window.
- ___ 10 If the information being entered is a fixed length, the text field is the same length as the information.
- ___ 11 If length of information varies, the text field is as long as the longest information.
- ___ 12 The text field includes scroll bars if the text being entered is longer than the field or extends beyond a single line.
- ___ 13 Strings longer than 5-7 characters long are entered in smaller chunks.
- ___ 14 The field format is meaningful to users and consistent with their expectations.
- ___ 15 If dynamic noneditable text is displayed in a text field, the field has a different appearance (e.g., background color) than an editable text field.
- ___ 16 When the pointer is on a noneditable text field, its shape does not change to an I-beam.
- ___ 17 A noneditable text field is not selectable; i.e., clicking on it does not change its appearance or display a text cursor in the field.

6.5.2 Behavior

6.5.2.1 Supporting Text Entry and Manipulation

- _____ 1 Variable-length text is automatically justified during text entry.
- _____ 2 Text characters are displayed as typed by users unless a special format is required
- _____ 3 When the application presents stored text in a text field, it is displayed in a consistent format so users recognize it as such, and any text editing by users is converted into this format.
- _____ 4 Numeric data can be entered from the keyboard or the numeric keypad.
- _____ 5 The amount of data users have to enter in a text field is minimized; automatic entry of data into a text field is performed whenever possible.
- _____ 6 Windows contain an indicator when position hooking to fill a text field is available.
- _____ 7 Users are not required to enter data in a mandatory field before moving to another field.
- _____ 8 Users accept the default value in a text field by tabbing to next field in window; tabbing does not affect default.
- _____ 9 If users modify the default but do not save the change, the change does not affect the default when the window containing the field is displayed again.
- _____ 10 Autotabbing is available only when data are broken into smaller groups of characters, with each group entered in a separate text field.

6.5.2.2 Error Checking and Correction

- _____ 1 The application performs error checking on the data entered and provides feedback when errors are detected.
- _____ 2 Users can fix errors by editing individual characters in the field, rather than having to erase and retype the entire field.
- _____ 3 When users enter data in fields that are unrelated to one another, they receive feedback about an error when they tab out of the text field but they are not prevented from leaving the field.
- _____ 4 Error feedback is provided in the message bar of the window if one is present; otherwise, feedback is provided by changing the appearance of the field with the error and/or in a separate message window.
- _____ 5 When users enter data in fields that are interdependent, they receive feedback concerning the errors when they attempt to enter or save the data.
- _____ 6 Users can save the data they have entered at any time by executing an explicit action.
- _____ 7 The application informs users if any errors are present and does not accept (i.e., save) the data until all errors are corrected.
- _____ 8 The application provides feedback to indicate the data have been saved.
- _____ 9 When users enter data, all are saved, regardless of text cursor position in window.

6.6 LIST BOXES

6.6.1 Appearance

- _____ 1 A list is used to select one or more items from a set of items.

- ___ 2 The items in a list are displayed vertically, with one item per line.
- ___ 3 A vertical scroll bar appears to the right of the list when the number of items exceeds the space available.
- ___ 4 The title of the list describes its purpose or contents, appears above the list box, and is not followed by a colon.
- ___ 5 Windows: If a list is inactive (i.e., unavailable), its title is dimmed.
- ___ 6 Windows: If an item in a list is unavailable for selection, it is omitted from the list (rather than displayed as unavailable).
- ___ 7 Windows: If scrolling is not possible in a list, the stepper arrows are dimmed, the slider is removed, and the color of the trough region changes to that of the window background.
- ___ 8 A list is large enough to display 6-8 items at a time, or all of the items if there are fewer than 6.
- ___ 9 If the items are similar in length, the list is wide enough to read the items without having to scroll horizontally.
- ___ 10 If the items differ significantly in length, the list is wide enough to read the average-length items and has horizontal scroll bar.
- ___ 11 If the items in a list represent possible attribute values for a selection, the current value is selected when the list is first displayed; if the selection is heterogeneous, none of the items is selected.
- ___ 12 The items in the list appear in sequential order based on nature of items and sequence expected (e.g., chronological, alphabetical, sequential, functional, by importance).
- ___ 13 Items added to a list appear in their correct position in the list, not at the end of the list.
- ___ 14 Selecting an item does not affect the order of the items in the list.
- ___ 15 A list scrolls only in response to user action and does not scroll automatically.

6.6.1.1 Drop-Down List Boxes (Windows Only)

- ___ 1 A drop-down list is used instead of a regular list when the space available in a window is limited.
- ___ 2 A drop-down list consists of a noneditable text area showing the currently selected item, a down-pointing arrow button, and a list of items that is displayed only when opened by the user.
- ___ 3 The same guidelines concerning height and width apply to an open drop-down list as to a regular list.
- ___ 4 The arrow button abuts the text area in the drop-down list.
- ___ 5 When users select an item in the list, it is displayed in the field and the list closes.

6.6.1.2 Multiple Selection List Boxes (Windows Only)

- ___ 1 A multiple selection list box is used when users need to select multiple discontinuous items.
- ___ 2 A multiple selection list includes a check button preceding each item.
- ___ 3 When users select an item in a multiple selection list, an X appears in the check button for that item.

6.6.2 Behavior

6.6.2.1 Navigation and Selection

- _____ 1 Placing the pointer on a list item and clicking BSelect moves the location cursor to the item and selects it.
- _____ 2 If other pointing device selection methods are available in the list, they are performed as defined in table 3-1.
- _____ 3 If the window containing the list has a default action, double clicking on an item selects the item and executes the action.
- _____ 4 <Up> and <Down> move the location cursor to the previous and next items in the list; <Left> and <Right> scroll the list one character to the left and right.
- _____ 5 <PageUp> and <PageDown> move the location cursor to the item one page up and down in the list; <Ctrl><PageUp> and <Ctrl><PageDown> scroll the list one page to the left and right.
- _____ 6 <Home> and <End> scroll to the leftmost and rightmost edge of the list; <Ctrl><Home> and <Ctrl><End> move the location cursor to the first and last items in the list.
- _____ 7 Motif: Selection from the keyboard is performed as defined in table 3-2.
- _____ 8 Windows: Selection from the keyboard is performed as defined in table 3-3.

6.6.2.2 Speed Search in Lists

- _____ 1 Speed search is available in lists containing more than 10 items.
- _____ 2 Incremental search is available in lists containing more than 50 items.
- _____ 3 Speed search and incremental search are not case-sensitive; if the search has to be case-sensitive, then this information is provided to users.
- _____ 4 Feedback (e.g., an auditory signal) is provided when no match is found in a speed search or incremental search.

6.7 SCROLL BARS

6.7.1 Appearance

- _____ 1 A scroll bar is used to view information when it exceeds the space available to display it.
- _____ 2 Vertical scroll bars control backward and forward movement through the information; horizontal scroll bars control left and right movement.
- _____ 3 A scroll bar contains a trough region, a slider, and stepper arrows.
- _____ 4 Relative slider position indicates the relative position of the information displayed in window.
- _____ 5 Users are able to scroll to the top or the bottom of the information but not beyond.
- _____ 6 Windows: When a window cannot be scrolled any further, the associated stepper arrow is dimmed.

6.7.2 Behavior

- _____ 1 Pressing BSelect on a stepper arrow moves the slider one unit (i.e., a column or a line) in the arrow direction.
- _____ 2 Pressing BSelect on a trough moves the slider one page length or width minus one unit in the direction indicated.
- _____ 3 Dragging the slider with BSelect moves the slider in the pointer direction.
- _____ 4 Dragging BTransfer on the trough moves the slider to the pointer position and then moves the slider in the pointer direction.
- _____ 5 <Esc> (or <Cancel> in Motif) returns the slider to its position before the sliding operation began.
- _____ 6 When the scroll bar has focus, the arrow keys move the slider one unit in the arrow direction.
- _____ 7 <Ctrl> in combination with the arrow keys move the slider one large unit in the arrow direction.
- _____ 8 <PageUp>, <PageDown>, <Ctrl><PageUp>, and <Ctrl><PageDown> page the scrollable area in the specified direction (less one unit of overlap).
- _____ 9 <Home> or <Ctrl><Home> and <End> or <Ctrl><End> scroll to the beginning and end of the scrollable region.

6.8 SCALES AND GAUGES

6.8.1 Appearance

- _____ 1 A scale is used to select a value in a range.
- _____ 2 A scale contains a trough region, a slider for selecting a value, and a label showing the current scale value; arrow buttons may also be included for selecting a value.
- _____ 3 The trough contains tick marks and is labeled with the minimum/maximum scale values.
- _____ 4 A gauge is used to display values that cannot be changed.
- _____ 5 If a gauge is used to indicate processing, the trough region fills dynamically to indicate the relative amount of processing completed.
- _____ 6 If the exact percentage of processing completed is important, the gauge is labeled to indicate the units of measurement represented by the gauge.
- _____ 7 A gauge contains a trough region but no slider or stepper arrows.
- _____ 8 A gauge can receive keyboard focus so users can access Help for that control.

6.8.2 Behavior

- _____ 1 Pressing BSelect on a stepper arrow moves the slider one unit in the direction indicated.
- _____ 2 Pressing BSelect on the trough moves the slider one large increment in direction indicated.
- _____ 3 Dragging the slider with BSelect moves the slider in the pointer direction.
- _____ 4 Dragging BTransfer on the trough moves the slider to the pointer position and then moves the slider in the pointer direction.
- _____ 5 <Esc> (or <Cancel> in Motif) returns the slider to its position before the sliding operation began.
- _____ 6 When the scale has focus, the arrow keys move the slider one unit in the arrow direction.

- ___ 7 <Ctrl> in combination with the arrow keys move the slider one large unit in the arrow direction.
- ___ 8 <Home> or <Ctrl><Home> and <End> or <Ctrl><End> move slider to the minimum and maximum scale values.

6.9 SEPARATORS

- ___ 1 A separator is used to delimit the elements in a menu or a window.
- ___ 2 A separator does not support internal navigation.

6.10 COMBINATION CONTROLS

6.10.1 Combo Boxes

- ___ 1 A combo box contains a text field and a list box below the text field.
- ___ 2 A drop-down combo box contains a text field, an arrow button, and a list box that is displayed when the button is depressed.
- ___ 3 Users select an item from the list to display in the text field or type directly in the field.
- ___ 4 When users select an item from the list, it replaces any text in the field.
- ___ 5 The text entered in the text field does not have to match an item in the list.
- ___ 6 Windows: As users type in the text field, the list scrolls to the first item that begins with the characters being typed.
- ___ 7 The list is large enough to display 6-8 items at a time, or all of the items if fewer than 6.
- ___ 8 A vertical scroll bar is provided when the list is too long to view all of the items.
- ___ 9 In a list that can be scrolled, the location cursor does not wrap.
- ___ 10 The combo box is wide enough for users to read all of the items in the list, with the text field the same width as the list.
- ___ 11 List items appear in sequential order based on nature of items and sequence expected.
- ___ 12 If it is appropriate for users to make no choice in a combo box, the list includes a “No Choice” item that, when selected, removes any text that is displayed in the text field.
- ___ 13 When displayed, the text field can either be empty or pre-filled with default list item.
- ___ 14 The default entry is highlighted when text field has focus so typing overwrites the entry.

6.10.2 Spin Buttons

- ___ 1 A spin button is used when users have to enter no more than 20 discrete, ordered values.
- ___ 2 A spin button contains a single-line text field and up/down arrow buttons to the right of the field.
- ___ 3 The text field is editable (if all values are not included) or noneditable (if all values are included).
- ___ 4 When a spin button is displayed, the text field contains a default value.
- ___ 5 Users click on the arrow buttons to step through the value in the text field.
- ___ 6 Users press on the arrow buttons to step through the values continuously in the direction of the arrow button.
- ___ 7 When largest or smallest value is reached, the entries wrap to cycle continuously through the full range.

- _____ 8 If the text field is editable, users can type a value directly in the field.
- _____ 9 If used for DTG or lat/long, separate spin buttons are provided for each part of the entry.
- _____ 10 Spin buttons can be combined with standard text fields for data entry.
- _____ 11 If a value is typed in the text field, it is validated for correct syntax and format when users move focus out of the field.

6.11 STANDARD AND NONSTANDARD CONTROLS

- _____ 1 If a nonstandard control is used, it has as much of standard “look and feel” as possible (i.e., a 3-D appearance and similar color and shape coding).
- _____ 2 A control that is a composite of existing controls has the appearance and behavior of the component controls.
- _____ 3 All controls in the application are identifiable solely on the basis of their appearance.
- _____ 4 All controls with the same function have the same appearance.
- _____ 5 Controls that are similar in shape have distinctive visual cues.
- _____ 6 COTS software is configured to comply with the specifications in this style guide.

7.0 SYSTEM AND APPLICATION DESIGN

7.1 CDE IMPLEMENTATION IN THE DII (Motif Only)

7.1.1 The Desktop in DII Systems

- _____ 1 The system provides access to desktop management functions in a CDE-compliant manner except where the specifications presented here indicate otherwise.
- _____ 2 Each application determines the constraints for movement and deletion of objects it creates, with users allowed to perform these actions if they are granted permission to do so.
- _____ 3 If these actions are allowed, they are performed using the functionality provided by the desktop. If these actions are not allowed, the corresponding desktop features are unavailable when the application is selected; applications do not alter how these functions are performed or remove them from the desktop.
- _____ 4 The system provides a classification bar with markings indicating the current classification level.
- _____ 5 The classification bar extends across the top of the screen and cannot be obscured by application windows when they are displayed or repositioned on the screen.
- _____ 6 The classification bar displays the current classification level in the middle of the bar and may include status indicators at the left margin and a digital clock at the right margin.
- _____ 7 The classification bar uses the colors listed in appendix D.
- _____ 8 The text in the classification bar is presented in upper-case letters, with all classification terms are spelled out and no embedded spaces within words in the label.

7.1.2 Session Management

- _____ 1 The system implements a login procedure that all users must complete before they can access any system functions.
- _____ 2 The system does not allow users to access a Terminal window unless this privilege has been granted to them.
- _____ 3 The system removes the Failsafe Session and the Command Line Login features in CDE.
- _____ 4 The system does not provide options to select a language other than English unless other languages are supported.
- _____ 5 The system supports both “current” and “home” session options and allows users to select between them.

7.1.3 Application Management

- _____ 1 The system provides access to the built-in application groups in Application Manager.
- _____ 2 Users launch an application either by double clicking on its application icon in Application Manager or pressing <Return> twice when focus is on the icon.
- _____ 3 If an application uses or creates data files, users launch it when they double click on one of its file icons in File Manager or when they drop the file icon on the application icon; in this case, the data file is loaded into the application window that is opened.
- _____ 4 If an application is available in the Personal Applications subpanel in the Front Panel, users launch it by clicking on its control in the subpanel or by dragging one of its file icons from File Manager to the control.
- _____ 5 Users cannot rename or delete an application icon unless they have been granted permission to do so.
- _____ 6 Users can launch only one instance of an application; double clicking on the icon for an application that is running only raises the application window to the front of the screen.
- _____ 7 Users cannot access the Terminal Emulator application in Application Manager or launch the Terminal application in the Front Panel or from the File Menu in File Manager unless they have permission to do so.
- _____ 8 Application Manager is the single repository for applications and application groups that the system adds to the desktop; DII applications are not installed in the Front Panel or its subpanels.
- _____ 9 Only those applications which users have permission to execute are available in Application Manager; applications (and any files or folders used by the application) to which users do not have access are not visible on the desktop.
- _____ 10 Each application is represented by a single application icon.
- _____ 11 The label for the icon is the name of the application.
- _____ 12 Users can access all of the functionality in the application from the application icon.

7.1.4 File Management

- _____ 1 The system uses File Manager to display the files and folders available to users.
- _____ 2 The system limits Terminal access from File Manager to those users with this privilege.

7.1.5 Workspace Management

- _____ 1 Users can add, remove, and rename workspaces, place applications in one, some, or all available workspaces, and switch between workspaces.
- _____ 2 A Workspace menu containing options for managing the workspace is available on all workspaces.
- _____ 3 If any of the following workspace management functions are included in a Workspace menu, they are ordered: Shuffle Up, Shuffle Down, Refresh, Minimize/Restore Front Panel, Restart Workspace Manager, and Log Out.
- _____ 4 The options in the Workspace menu execute the actions listed in appendix C.
- _____ 5 The system implements Graphical Workspace Manager, provides an Application List option in the Workspace menu, and adds a Workspaces control to Style Manager.
- _____ 6 If the system uses multiple monitors, it configures the desktop to display a separate Front Panel on each screen, with each screen functioning as a unique desktop environment.

7.1.6 Style Management

- _____ 1 The system makes the standard color, font, backdrop, keyboard, mouse, beep, windows, and startup controls available to users in Style Manager.
- _____ 2 The system supports the Screen With Lock extension to the Screen control so that users can choose to enable or disable a screen saver and screen lock on timeout.
- _____ 3 If the system is installed in an office-like environment, it implements the color palettes listed in appendix D.
- _____ 4 If the system is installed in another environment, it defines system-specific palettes and makes them available to users in the Color control in Style Manager.
- _____ 5 The system implements the centralized color management capability provided by the desktop so that applications change color dynamically when users select a different palette.
- _____ 6 Applications that cannot use dynamic colors use the DII default palette as their color set.
- _____ 7 The system implements the centralized font management capability provided by the desktop, using the default fonts defined by the platform on which the system is installed.
- _____ 8 Applications use the fonts for which font aliases are defined in CDE.

7.2 APPLICATION DESIGN IN THE DII

7.2.1 Applications Providing Centralized Task Management

7.2.1.1 Basic Implementation

- _____ 1 Launching the application opens the window with application control.
- _____ 2 Primary windows have a Close menu option or a Close push button, both of which close all of the windows in the family parented by the window and quit processing in the window.
- _____ 3 An Exit option is available only from the menu bar of the application control window; selecting this option closes all windows and quits processing in the application.
- _____ 4 The title of the window with application control is the name of the application.
- _____ 5 Other primary windows are identified by application and task name.

- _____ 6 Primary windows in which a file has been loaded are identified by application name, followed by a hyphen, and the name of the file.
- _____ 7 Secondary windows are identified by application name, followed by a colon, and the name of the window.

7.2.1.2 Nested Implementation

- _____ 1 The parent segment provides application control for the overall application.
- _____ 2 The application icon on the desktop is named for the parent segment; double clicking on the icon launches this segment.
- _____ 3 Closing a primary window in the parent segment has no effect on windows in a child segment.
- _____ 4 Exiting the window with application control in the parent segment closes all of the windows and quits processing in both parent and child segments.
- _____ 5 The child segment is not available on the desktop and can only be launched from within the parent segment.
- _____ 6 A Close option is available from the menu bar of the primary window(s) in the child segment; selecting this option closes all windows in the family parented by the window and quits processing in the window.
- _____ 7 The name of the parent segment is used as the application name in the title of all primary windows in the child segment.

7.2.2 Applications Providing Distributed Task Management

7.2.2.1 Basic Implementation

- _____ 1 A pop-up menu is attached to the application icon listing all of the top-level tasks in the application.
- _____ 2 Double clicking on the application icon launches the default task and opens its primary window.
- _____ 3 Selecting one of the tasks from the pop-up menu launches the task, opens its primary window, and changes the menu option to unavailable.
- _____ 4 Each primary window includes menu options from which all of the other top-level tasks in the application can be accessed.
- _____ 5 Close and Exit options are available from the menu bar of each primary window in the application; selecting either of these options closes all windows in the family parented by the window and quits processing in the window.
- _____ 6 Exiting the last primary window in the application also exits the application.
- _____ 7 Primary windows in the application are identified by task name.
- _____ 8 Secondary task windows are identified by task name, followed by a colon, and then the name of the secondary window.

7.2.2.2 Resource Sharing Among Applications

- _____ 1 Each mission and support application is closed and exited independently; executing these actions in any of the applications affects only the windows in the application.

- _____ 2 When a support application is launched from within a mission application but is not shared, the latter application can provide an Exit All option to exit both it and the support application.
- _____ 3 When a support application is launched as a shared resource and one of the mission applications is exited, the shared window remains open but the data from the application being exited are removed from the window; an Exit All option is not available.

7.2 3 Other Design Considerations

7.2.3.1 Icon Design in CDE (Motif Only)

- _____ 1 The icons for the application have a common theme, with individual icons tailored to reflect specific application functions.
- _____ 2 Icon graphics have a three-dimensional style.
- _____ 3 The shape of a container icon provides cues as to its function and the kind of objects it contains.
- _____ 4 A document or file icon indicates what application is associated with the document or file and the kind of data stored in the icon.
- _____ 5 If an application supports multiple file formats, document icons use the same basic file graphic but have a different “tag” to distinguish each output format.

7.2.3.2 Accessing Segments Within an Application

- _____ 1 If menus in the application become excessively long, it uses submenus to reduce menu length.
- _____ 2 Menus that cannot be shortened include controls for viewing and selecting the options that extend beyond the bottom of the screen.
- _____ 3 The application determines when users can and cannot access each of the tasks performed by a segment and dims the associated menu option to indicate its unavailability.

8.0 APPLICATION WINDOW DESIGN

8.1 PRIMARY AND SECONDARY TASK WINDOWS

8.1.1 Window Components

- _____ 1 A primary task window contains a window frame with resize borders, Window menu, Minimize, and Maximize buttons, and a title bar.
- _____ 2 Motif: The Window menu in a primary task window contains (in this order) Restore, Move, Size, Minimize, Maximize, Lower, Occupy Workspace, Occupy All Workspaces, Unoccupy Workspace, and Close options.
- _____ 3 Windows: The Window menu in a primary task window contains (in this order) Restore, Move, Size, Minimize, Maximize, and Close options and may include Switch To and Next options.
- _____ 4 A primary task window includes a menu bar and may include a message bar.

- _____ 5 A secondary task window contains a window frame with resize borders, a Window menu, and a title bar; the window may also contain Minimize and Maximize buttons.
- _____ 6 Motif: The Window menu in a secondary task window contains (in this order) Restore, Move, Size, Minimize (if available), Maximize (if available), Lower, and Close options.
- _____ 7 Windows: The Window menu in a secondary task window contains (in this order) Restore, Move, Size, Minimize (if available), Maximize (if available), and Close options.
- _____ 8 A secondary task window includes a menu bar at the top or an action area at the bottom; the window may have a message bar.

8.1.2 Window Design Guidelines

8.1.2.1 Window Title

- _____ 1 The window title is centered in the title bar and presented in mixed case, with the first letter of each word capitalized.
- _____ 3 If a file name is included in the title, it is in mixed case in Motif and in upper case letters (including any extensions) in Windows.
- _____ 3 Each window title in the application is unique.
- _____ 4 The title does not contain version/path information and does not present information that changes dynamically.
- _____ 5 If selecting a menu option causes a secondary window to be displayed, the title of the window matches or refers to the wording of the option that displayed it.

8.1.2.2 Menu Bar

- _____ 1 If a window includes a menu bar, it appears below the title bar.
- _____ 2 The menu bar contains no more than ten menu titles plus Help.
- _____ 3 Motif: Menu titles begin at the left margin and extend rightward, with Help at the right margin.
- _____ 4 Windows: Menu titles begin at the left margin and extend rightward; Help is the last menu and placed next to the preceding menu title.
- _____ 5 The space between menu titles is sufficient (at least three character widths) so that multi-word titles can be distinguished from single-word titles.
- _____ 6 Commands (e.g., push buttons) are not included in a menu bar.

8.1.2.3 Common Menus

- _____ 1 Motif and Windows conventions concerning menu design and content are followed except as needed to provide access to application-specific functions.
- _____ 2 If any common menus are used, they are ordered: File, Edit, View, Options, Window (Windows only), Help.
- _____ 3 The first menu contains options for users to work with the data in the window as a whole; the title of this menu is File or an application-specific term with comparable meaning.

- _____ 4 Motif: If a File menu includes any of the following options, they are ordered: New, Open, Save, Save As, Print, Close, and Exit. Separators follow Open, Save As, and Print options.
- _____ 5 Windows: If the File menu includes any of the following options, they are ordered: New, Open, Close (if available), Save, Save As, Print, Print Setup, and Exit. Separators follow the Save As and Print Setup options.
- _____ 6 Windows: If the File menu includes a list of most recently used files, the list precedes the Exit option.
- _____ 7 Windows: Selecting an option with a file name opens a window containing the file; if the file is already open, selecting the option raises that window to the front.
- _____ 8 Windows: The number of files in the list ranges from three to eight but remains constant within the application.
- _____ 9 Windows: When a file is opened, the file name is placed at the top of the list in the File menu and given the number 1 which is used as its mnemonic; when another file is opened, it is added to the top of the list (and given the number 1) and the previously opened files move down in the list.
- _____ 10 If an Edit menu is present, it contains options for users to modify the data in the window.
- _____ 11 If File and Edit are both present, they are next to each other in the menu bar.
- _____ 12 Motif: If an Edit menu includes any of the following options, they are ordered: Undo, Cut, Copy, Copy Link, Paste, Paste Link, Clear, Delete, Select All, Deselect All, Select Pasted, Reselect, Promote. A separator follows the Undo, Paste Link, and Delete options.
- _____ 13 Windows: If an Edit menu includes any of the following options, they are ordered: Undo, Repeat, Cut, Copy, Paste, Paste Special, Clear, Delete, Select All, Find, Replace, and Links. Separators follow the Repeat, Select All, and Replace options.
- _____ 14 If a View menu is present, it contains options for changing the user's view of the data but does not actually change the data.
- _____ 15 If an Options menu is present, it contains options for customizing the application.
- _____ 16 Windows: If a Window menu is present, it contains options for manipulating document windows.
- _____ 17 Windows: The Window menu includes a New Window option, followed by window arrangement commands, and finally a list of open windows.
- _____ 18 Windows: Selecting a window name from the Window menu raises that document window to the front and gives it focus.
- _____ 19 Windows: The active window is indicated by a check mark preceding the window name in the Window menu.
- _____ 20 Windows: The Window menu contains up to nine window names; if more than nine windows are open, the menu includes a More option that displays a dialog window with the names of all open document windows.
- _____ 21 The Help menu provides access to additional information about the window or the application.
- _____ 22 Motif: If the Help menu includes any of the following options, they are ordered: Overview, Index, Table of Contents, Tasks, Reference, Tutorial, Keyboard, Mouse, Mouse and Keyboard, On Item, Using Help, and About.

- ____ 23 Motif: The minimum set of options in a Help menu is Overview, Tasks, Reference, On Item, Using Help, and About <application> options and may include a Keyboard option.
- ____ 24 Windows: If the Help menu includes any of the following options, they are ordered: Contents, Search for Help On, Tutorial, How to Use Help, and About.

8.1.2.4 Arrangement of Controls

- ____ 1 Controls performing a similar/related function are grouped together and surrounded by a frame.
- ____ 2 If a heading is included, it is a label describing the function performed by the controls in the group.
- ____ 3 The heading is placed inside the frame or in the frame.
- ____ 4 If the heading is longer than the text in controls, the frame size is extended to be wider than the heading.
- ____ 5 The heading is either left justified or centered within the frame.
- ____ 6 The heading is in mixed case, following normal capitalization rules, and not followed by a colon.
- ____ 7 The preferred orientation for a group of radio or check buttons is vertical and left-aligned.
- ____ 8 If a group of radio or check buttons is placed horizontally, space is sufficient (at least twice the distance between the button and its label) so the button is paired with the label on the right, not left.
- ____ 9 When a window is displayed, all of the controls reflect current state of the application.
- ____ 10 If there is an expected choice in a group of controls, it is selected (i.e., highlighted) when the window is initially displayed.
- ____ 11 Controls that are temporarily unavailable are dimmed and not available for selection.
- ____ 12 Controls that are never available to users do not appear in a window.
- ____ 13 When a control is selected but not executed, the selection is not saved and the control reverts to its original state when the window was first displayed.

8.1.2.5 Availability of Scroll Bars and Window Panes

- ____ 1 If a scrollable area is sized too small to view all of the contents, scroll bars are provided to allow scrolling of the area.
- ____ 2 If scroll bars are needed, they are located to the right or at the bottom of the area being scrolled.
- ____ 3 Scroll bars scroll the main part of window only and not the menu bar or message bar in the window.
- ____ 4 Motif: A scrollable window always displays its scroll bars regardless of whether the viewable window is the same size or smaller than the underlying data.
- ____ 5 Windows: Scroll bars remain displayed even if they become inactive.
- ____ 6 Window panes are used to separate control areas when space is limited or to present two simultaneous views of the same data in a single window.
- ____ 7 Motif: Users resize panes by dragging the boundary between the panes using BSelect or BTransfer or by moving the sash using the arrow keys.

- ___ 8 Motif: Making one pane larger makes the other pane smaller but not affect the overall size of the window.
- ___ 9 Windows: Users drag the split box to divide the window into separate panes; double clicking on the split box divides a window in the middle.
- ___ 10 Windows: Dragging the split box or split bar to either end of the window closes the pane in the direction of the drag.
- ___ 11 Windows: When a window is split, scroll bars are displayed so that users can scroll each pane independently within the window.

8.1.2.6 Arrangement of Push Buttons

- ___ 1 Motif: Push buttons are displayed horizontally, centered at bottom of the window, and separated from other controls with a separator.
- ___ 2 Motif: Push buttons are ordered left to right based on sequence of use, with the one most often used on the left.
- ___ 3 Motif: Buttons for positive actions are on the left, followed by buttons for negative and canceling actions.
- ___ 4 Motif: Every window includes a Help push button which is the rightmost button.
- ___ 5 Windows: Push buttons are placed across the bottom of a window.
- ___ 6 Windows: Buttons that initiate actions are placed at the left, followed by “GoTo,” “GoSub,” and Help buttons.
- ___ 7 Windows: If there is an OK button, it is placed first and followed by Cancel, with both separated from the other buttons; if there is no OK button, Cancel follows the other action buttons but precedes the “GoTo,” “GoSub,” and Help buttons.
- ___ 8 Push buttons appear in the same order throughout the application.
- ___ 9 The window contains no more than 7 push buttons, including Help.
- ___ 10 Close and Cancel are not included as push buttons in the same window.
- ___ 11 Windows: If the actions performed in a window make irreversible changes to data, the label of the Cancel button changes to Close as soon as the first such action is executed. When the window is closed and then reopened, the button label reverts to Cancel.
- ___ 12 If push button actions affect different objects in the window, their labels reflect what each affects and they are placed near the object(s) to which they relate.
- ___ 13 Motif: Separate push buttons are provided for mutually exclusive actions; the one that is unavailable is grayed.
- ___ 14 Windows: If the function performed by a push button changes depending on the state of the application, the label changes accordingly; a single push button, rather than separate buttons, is used in the window.
- ___ 15 Windows: If a “GoTo” or “GoSub” window can be opened as a result of a menu selection, the window title is the same as the menu option.
- ___ 16 Windows: If users open a “GoSub” window, make changes in the window that cannot be undone, and then return to the parent window, selecting Cancel in the parent cancels all changes made in both the parent and child windows.

8.1.2.7 Default Push Buttons

- ___ 1 If a default action is available in the window, it is the action that users are most likely to execute in the window.
- ___ 2 If an expected choice cannot be anticipated, there is no default defined for the window.
- ___ 3 If there is no default button in a window, <Enter> (or <Return> in Motif) has no effect; users have to select one of the available push buttons to execute an action.
- ___ 4 Motif: When a window is initially displayed, the default push button is the leftmost button in a group of push buttons.
- ___ 5 The same push button is the default whenever the window is displayed.
- ___ 6 When focus is on a push button, its action is the default and it is shown with default highlighting.
- ___ 7 The default highlight moves with the location cursor during keyboard navigation in a group of push buttons and returns to the original button when focus leaves the push button group.
- ___ 8 If the default action in a window varies, one push button always shows the default highlighting except when there is no default action currently available.
- ___ 9 If focus is outside a window, the default highlight is displayed on the push button whose action will be the default when focus returns to the window.
- ___ 10 When more than one action is available in a window, the default push button is the nondestructive one.
- ___ 11 If the default highlight moves to a push button other than the one originally identified as the default, the highlight reverts to the original button when the window is closed and then opened again.
- ___ 12 The action performed by the default push button is reversible.

8.1.2.8 Tool Bars

- ___ 1 If a tool bar is included in a window, it provides redundant access to actions and settings available elsewhere in the window.
- ___ 2 Motif: A tool bar is used only in windows with a menu bar; if present, the tool bar is located at the top of the window, below the menu bar.
- ___ 3 Motif: The tool bar is the same width as the window and the same height as the menu bar.
- ___ 4 Windows: If a tool bar is placed in a separate dialog window, it is always displayed in front of the window to which it applies.
- ___ 5 Windows: The dialog window with the tool bar includes a title bar and a Window menu with Move and Close options.
- ___ 6 A window with a tool bar includes a message bar so that information about individual buttons can be provided to users.
- ___ 7 Users can hide or show a tool bar.
- ___ 8 Motif: A tool bar contains no more than 20 buttons; the buttons are of equal size and evenly spaced across the tool bar.
- ___ 9 Motif: The buttons in a tool bar are arranged in an order expected by users or by frequency or sequence of use or importance.
- ___ 10 A toolbox (i.e., a toolbar placed along the left margin of a window) contains groups of action buttons arranged vertically, with no space between the buttons.

- ___ 11 The normal appearance of a button is raised; when selected, the button is recessed and changes appearance to indicate its selected state.
- ___ 12 A button that becomes unavailable changes appearance (e.g., is grayed out) to show it cannot be selected.
- ___ 13 Whenever a menu option becomes unavailable, the corresponding button in the tool bar is also shown as unavailable.
- ___ 14 The icon graphics are the same size in each button in the toolbar.
- ___ 15 Windows: The graphic image is a black outline, with white fill within the outline where needed.
- ___ 16 The icons depict a before/after representation, the tool to perform the action, or the action itself.
- ___ 17 Windows: If text is included with the graphic, button size is larger and dependent on text length.
- ___ 18 Windows: If text is included with the graphic, the graphic is placed either above the text or to the left of the text.
- ___ 19 A button remains selected as long as the mode invoked by button is in effect.
- ___ 20 The pointer shape changes to indicate type of operation users can perform in the mode.
- ___ 21 The pointer retains its modified shape whenever it is in window where the mode is in effect.
- ___ 22 If the pointer is outside the window with the mode, it changes to the appropriate shape.
- ___ 23 A tool bar either provides a button for returning to an unselected state or automatically returns to an unselected state after an action is executed.

8.1.2.9 Message Bar

- ___ 1 If a window includes a message bar, it presents noncritical application messages to users.
- ___ 2 When the message bar is used to indicate status, a progress message is displayed when the action is initiated and updated when the action is completed; the text is removed from the message bar within 5 sec of action completion.
- ___ 3 A message bar contains read-only text; users cannot type or modify any text in this area.
- ___ 4 Motif: The margin area at the bottom of the window is widened so that messages can be presented there.
- ___ 5 Windows: If a status bar is used in a window, the application provides users with a way to hide or show the bar as desired.

8.1.2.10 Draggable Objects in Windows (Motif Only)

- ___ 1 An icon is included in a window to indicate that it contains a draggable object.
- ___ 2 The icon graphic is the same as the one used to represent the object in the File Manager on the desktop.
- ___ 3 The icon is placed next to any display of the contents of the object if one is present in the window, or in the upper right corner if there is no such display.
- ___ 4 The icon includes a label describing the kind of object the icon graphic represents.
- ___ 5 The icon graphic is used as the source indicator in the drag icon.

8.1.2.11 Pop-up Menus and Text Fields in Windows

- _____ 1 Pop-up menus are provided for those elements in a window for which redundant access to frequently executed functions would improve task performance.
- _____ 2 Windows: Read-only pop-up text fields are used to display additional information about the text in a field when space within a window is limited.
- _____ 3 Windows: The presence of a pop-up field is indicated by underlining the word(s) in the text to which it relates.
- _____ 4 Windows: Clicking BSelect on the underlined text displays the pop-up field; clicking anywhere outside the field dismisses it.
- _____ 5 Windows: When the field is displayed, it is placed so that its top left corner is at the same position as the top left corner of the original text.

8.1.2.12 Mnemonics and Accelerators in Windows

- _____ 1 Windows: Mnemonics are available as an additional method for keyboard navigation among controls in a window.
- _____ 2 Windows: When mnemonics are implemented in a window, they behave as indicated in section 3.4.2 and 5.5 and use the characters listed in appendix C.
- _____ 3 Windows: The OK and Cancel push buttons do not have mnemonics.
- _____ 4 Accelerators are not assigned to individual controls in windows.

8.1.2.13 Document Windows (Windows Only)

- _____ 1 A document window has the same window components as a primary window.
- _____ 2 The Window menu in a document window contains the same options as this menu in the application window.
- _____ 3 The title of a document window is the name of the document and presented in mixed case.
- _____ 4 Document windows appear within the borders of the application window.
- _____ 5 When a document window is maximized, the window is closed; the data are displayed in the application window, its title changes to include the document name, the Window menu button from the document window and a Restore button are added to its menu bar, and a scroll bar is displayed if the document requires scrolling.

8.2 DIALOG WINDOWS

8.2.1 Window Components

- _____ 1 A dialog window contains a window frame, a Window menu button, and a title bar; the window does not have resize borders or Minimize or Maximize buttons.
- _____ 2 The Window menu in a dialog window contains (in this order) Move and Close options.
- _____ 3 Windows: If a dialog window is not movable, it does not have a title bar, and Move is not an option in the Window menu.
- _____ 5 The main area includes a control area for presenting messages or controls and a push button area at the bottom of the window for executing actions.

- ___ 6 Motif: Message windows are modeless whenever possible; an error message window is modal only if it is critical that users acknowledge having read the message prior to continuing to interact with the application.
- ___ 7 Windows: Critical message windows are system modal while information and warning message windows are application modal.
- ___ 8 Windows: The frame of a modal window has a colored inner border; the frame of a modeless window does not.

8.2.2 Window Design Guidelines

- ___ 1 Motif: The title of a dialog window includes the name of the application and describes the purpose of the dialog window.
- ___ 2 Motif: A dialog window contains a separator between the control area and action area of the window.
- ___ 3 Windows: The title of a message window is the application name and does not include the word "Error."
- ___ 4 If a dialog window includes a text message, it uses language that is meaningful to users and requires no further documentation or translation.
- ___ 5 The text in a message is left justified within the window.
- ___ 6 In messages with more than one sentence, the important information is placed at the start of the message.
- ___ 7 The message is worded so that the action required appears as a push button in the window.
- ___ 8 The application does not use a timed-information window, then resume processing on its own.
- ___ 9 A dialog window contains at least one push button that either performs the dialog window action and dismisses it (e.g., OK) or dismisses the window without taking any action (e.g., Cancel).
- ___ 10 A default push button is available in each dialog window in the application.
- ___ 11 If an action executed in a dialog window results in an error that generates an error message window, the dialog window remains displayed while the error window is presented and then dismissed.
- ___ 12 Push button order in modal dialog windows is OK/Cancel/Help, with OK as the default.
- ___ 13 Push button order in modeless dialog windows is OK/Apply/Cancel/Help or OK/Apply/Reset/Cancel/Help; OK is the default in windows that perform single actions and Apply is the default in windows that perform multiple actions.
- ___ 14 <Esc> (or <Cancel> in Motif) has the same effect as selecting the Cancel push button in the window.
- ___ 15 Auditory feedback accompanies message windows containing critical information.
- ___ 16 Users can set the level of auditory feedback or disable it temporarily as needed.

8.2.3 Message Dialogs (Motif Only)

8.2.3.1 Error Message Windows

- ___ 1 An error window is displayed to inform users when an error occurs.

- _____ 2 An error window includes the error symbol, a text message, and OK/Help or Continue/Cancel/Help push buttons.
- _____ 3 The message in an error window describes the error, why it happened, and what should be done to correct it.

8.2.3.2 Information Message Windows

- _____ 1 An information window is displayed to convey noncritical information to users.
- _____ 2 An information window includes the information symbol, a text message, and OK or OK/Help push buttons.

8.2.3.3 Question Message Windows

- _____ 1 A question window is displayed to request clarification of a previous user response.
- _____ 2 A question window includes the question symbol, a text message, and Yes/No/Help push buttons.

8.2.3.4 Warning Message Windows

- _____ 1 A warning window is displayed to convey critical information on user actions and to allow users to cancel a destructive action.
- _____ 2 A warning window includes the warning symbol, a text message, and Yes/No/Help or Continue/Cancel/Help push buttons.
- _____ 3 An audio signal accompanies the window to alert users to the warning.

8.2.3.5 Working Message Windows

- _____ 1 A working window is displayed to inform users when processing time exceeds 10 sec or when users need to cancel an operation in progress.
- _____ 2 A working window includes the working symbol, a text message, and an OK push button, and may include Cancel and Stop push buttons.
- _____ 3 Cancel interrupts the operation and returns the application and data to its state before the operation was activated.
- _____ 4 Stop interrupts the operation but does not reverse any changes already caused by the operation.
- _____ 5 During lengthy processing (in excess of 1 min), a working window is updated to indicate status of processing if update information is available.
- _____ 6 A working window remains displayed until the action is complete, the window doing action is minimized, or the user selects Cancel.
- _____ 7 When processing is complete, the working window is removed (without user action).
- _____ 8 Users can cancel the operation in progress, with confirmation required if unsaved data will be lost.

8.2.4 Message Dialogs (Windows Only)

8.2.4.1 Information Message Windows

- _____ 1 An information message window is displayed to provide information about the results of commands.
- _____ 2 An information message window contains the information symbol, a text message, and OK or OK/Help push buttons.

8.2.4.2 Warning Message Windows

- _____ 1 A warning message window is displayed to present error information or to allow users to cancel a destructive action.
- _____ 2 A warning window contains the warning symbol, a text message, and a push button for each choice available in the window plus a Help button.
- _____ 3 If the message text is worded as a question, the window includes Yes/No/Help push buttons.

8.2.4.3 Critical Message Windows

- _____ 1 A critical message window is displayed to present messages that must be corrected before users can continue to work in the application.
- _____ 2 The window contains the critical symbol, a text message, and push buttons for each choice available in the window plus Help.

8.2.5 Selection Dialogs (Motif Only)

8.2.5.1 Command Windows

- _____ 1 A command window is displayed when users need to enter keyboard commands.
- _____ 2 A command window includes a list displaying the command history and a text field for entering new commands but has no push buttons.
- _____ 3 The list box includes a vertical scroll bar when the command history exceeds the visible area in the list.
- _____ 4 The command history is cleared whenever the application is exited, and is resumed when the application is launched again.
- _____ 5 The text field is wide enough for users to view and read an entire command; a horizontal scroll bar is not included unless command lines are unusually long.
- _____ 6 Selecting an item from the list displays it in the text field.
- _____ 7 <Enter> or <Return> executes a command and adds it to the bottom of command history list.
- _____ 8 <Tab> moves the location cursor between the list and the text field.
- _____ 9 When focus is on the text field, <Up>, <Down>, <Ctrl><Home>, and <Ctrl><End> move the location cursor among items in the list and change the contents of the text field.

8.2.5.2 Prompt Windows

- _____ 1 A prompt window is displayed to request information needed to continue processing.

- _____ 2 A prompt window includes a message stating what information is needed, a text field for typing, and OK/Cancel/Help, OK/Apply/Cancel/Help, or OK/Apply/Reset/Cancel/Help push buttons.
- _____ 3 The text field has keyboard focus when a prompt window is initially displayed.

8.2.5.3 Selection Windows

- _____ 1 A selection window is displayed when users need to make a selection from a list of choices.
- _____ 2 A selection window contains a list box displaying the choices available, a text field for entering/displaying a choice, and OK/Apply/Cancel/Help push buttons.
- _____ 3 Both the list and text field include a heading that describes their contents.
- _____ 4 The list has a vertical scroll bar when number of items exceeds the visible area in the list.
- _____ 5 Selecting an item from the list displays it in the text field.
- _____ 6 If users type in the text field, the list scrolls to that item.
- _____ 7 If the text typed in the text field does not match any items in the list, users are prompted to add the item to the list.
- _____ 8 When users select OK or press <Enter> or <Return>, the selection is executed and the window closed.
- _____ 9 <Tab> moves the location cursor between the list and the text field.
- _____ 10 When focus is on the text field, <Up>, <Down>, <Ctrl><Home>, and <Ctrl><End> move the location cursor among items in the list and change the contents of the text field.

8.2.5.4 File Selection Windows

- _____ 1 A file selection window is displayed when users need to choose a file or directory.
- _____ 2 The window contains a text field for displaying and editing the current directory path, list boxes for displaying directory and file names, a text field for displaying and editing a file name, and OK/Update/Cancel/Help push buttons.
- _____ 3 When the window is used to specify an existing file, the OK push button is replaced with Open and it is the default action.
- _____ 4 When the window is used to specify a new file name, the OK push button is replaced with Save and it is the default action.
- _____ 5 The items in the Directory and File lists are presented in alphabetical order, with the first item in the Directory list the parent directory and labeled “..”.
- _____ 6 The window does not display hidden (i.e., dot) directories or files unless users need access to these types of files; if access is required, the window includes a check button to show or hide these files.
- _____ 7 The Directory text field presents the full path name; the File text field and the lists in the window shows relative path names.
- _____ 8 When users open the file selection window associated with a particular primary window, the directory location displayed is the default for that primary window.
- _____ 9 If users change the directory and then reopen the file selection window, the directory location is the one that was previously set by the user.

- ___10 When users close the primary window, the directory location in the file selection window reverts to the default for the primary window.
- ___11 If the application supports multiple primary windows, the directory reverts to the default defined for that primary window.
- ___12 When users open a file selection window, the File text field has keyboard focus.
- ___13 When users open a file selection window, the File list displays the contents of the current directory.
- ___14 The File list is updated when users edit the Directory text field and press <Enter> or <Return> or when they select a directory in the File list.
- ___15 When users select a file from the File list, the file name appears in the File text field.
- ___16 The application executes the selection(s) in the window when users select an item in the File list and activate the OK (or comparable) push button, when users double click BSelect on an item in the File list, or when users select a file name and press <Return> or <Enter> when the File text field has keyboard focus.
- ___17 Users are prompted to confirm the action executed in the window if the action will overwrite an existing file.

8.2.5.5 Print Dialog Windows

- ___ 1 A print dialog window is displayed when users need to select options for printing a file, a selection, or other type of object.
- ___ 2 The window contains a common area with standard information about the print job, an optional area with information specific to the application or function, and Print/Cancel/Help push buttons.
- ___ 3 The common area, located in the top part of the window, displays the name of the file or object type and includes controls for entering the printer destination, the number of copies desired, and the text to appear on the banner page, if any.
- ___ 4 The default entry in the Printer Destination control is the printer that is the default destination in the system; users can select or type any other valid printer name; the application saves the last user entry in this field and displays it when the window is opened again.
- ___ 5 The same information is presented in the common area in all print dialogs, with any application-specific controls placed in the optional area in the lower part of the window.
- ___ 6 If Reset and Print Preview actions are included in the window, they are available as push buttons and inserted between the Print and Cancel buttons.
- ___ 7 Separators are used between the common area, the optional area, and the push buttons.

8.2.6 Selection Dialogs (Windows Only)

8.2.6.1 File Open and File Save Windows

- ___ 1 The standard File Open window is used to choose a file or directory.
- ___ 2 The window contains a noneditable text area displaying the current position in the directory tree, list boxes for displaying directory and file names, a text field for displaying and editing a file name, drop-down lists for selecting drives and file types, and OK and Cancel push buttons.

- _____ 3 Users navigate in the window by either selecting from the Drives or Directories control or by typing this information in the File Name text field; when users switch to a new drive, the contents of the Directories list shows the contents of this drive, with the root directory at the top.
- _____ 4 Users can select a file by either typing in the File Name text field or selecting one of the items in the list box below the text field; if desired, users can filter the types of files included in this list by selecting a file type from the List Files of Type drop-down list.

8.6.2.2 Print Windows

- _____ 1 The standard Print window is used to submit a print job.
- _____ 2 The window identifies the printer to which the job will be sent and contains controls for specifying the print range, print quality, and number of copies and choosing to print to file or to collate the copies.
- _____ 3 These controls appear in the common part of the Print window, with application-specific controls available in an optional area in the lower part of the window.

9.0 INFORMATION PRESENTATION

9.1 TEXT INFORMATION

9.1.1 Text Font, Size, and Style

- _____ 1 Text is presented using the default font, size, and style defined by CDE or Windows.
- _____ 2 When a choice of fonts is available to the application, a sans serif bold font is used.
- _____ 3 Minimum text character height is 1/200th of the viewing distance.

9.1.2 Capitalization, Grammar, and Punctuation

- _____ 1 All text (including titles and major headings) is presented in mixed-case, following standard capitalization rules.
- _____ 2 Upper-case letters are used for acronyms and abbreviations and for emphasis in text.
- _____ 3 Arabic rather than Roman numerals are used when information has to be numbered.
- _____ 4 Continuous text is phrased in simple sentences, in the affirmative, and in active voice.
- _____ 5 A sequence of events or steps is presented in the order the steps are performed.
- _____ 6 The referent for “it” or “they” in a sentence is easily identified.
- _____ 7 Normal punctuation rules are followed, and contractions and hyphenation are avoided.
- _____ 8 Paragraphs are kept short and separated by at least one blank line.

9.1.3 Acronyms and Abbreviations

- _____ 1 Acronyms and abbreviations are used only if they are shorter than the full name and commonly understood by users.
- _____ 2 Abbreviations are the shortest possible length that will ensure uniqueness.
- _____ 3 Abbreviations are used consistently throughout the application.
- _____ 4 Words not commonly abbreviated are not abbreviated.

- _____ 5 Acronyms and abbreviations comply with relevant MIL-STDs.
- _____ 7 A dictionary is available (e.g., in Help) for decoding abbreviations/acronyms.

9.1.4 Formats for Date/Time and Latitude/Longitude

- _____ 1 The format for presenting date information is YYMMDD.
- _____ 2 The format for presenting time information is HHMM[SS]Z.
- _____ 3 The format for date/time group is DDHHMMZ MMM YY.
- _____ 4 Latitude/longitude information is displayed in separate fields, with Lat/Long labels.
- _____ 5 The format for latitude information is D{D}H or DD{MM{SS}}H.
- _____ 6 The format for longitude information is D{D{D}}H or DDD{MM{SS}}H.

9.1.5 Wild Card Characters in Text Searches

- _____ 1 Users can enter wild card characters to search for specific text patterns.
- _____ 2 @ searches for the occurrence of a single upper- or lower-case alphabetic character.
- _____ 3 # searches for the occurrence of a single numeric character.
- _____ 4 ? searches for the occurrence of a single alphanumeric character.
- _____ 5 * searches for the occurrence of zero or more alphanumeric characters.

9.1.6 Presenting Tabular Information

- _____ 1 Each column of tabular information has a heading and is clearly separated (by at least four character spaces) from information in other columns.
- _____ 2 Data groupings are indicated by blank space, separator lines, and/or different intensity levels; multiple colors are used only if they provide additional meaning.
- _____ 3 Alphabetic information is left-justified; integers are right-justified; decimal information is justified on the decimal point.
- _____ 4 Long strings of numbers are delimited with spaces or commas; leading zeros are not used.
- _____ 5 If the information extends beyond a line, additional lines are indented to indicate they are continuations.
- _____ 6 Tabular information is arranged in sequential, spatial, alphabetical, functional, or chronological order.
- _____ 7 Information that is important, requires immediate response, and/or is frequent is presented first in the table.

9.2 GRAPHICAL INFORMATION

9.2.1 Line Graphs and Surface Charts

- _____ 1 Line graphs are used to present trend, spatial, time-critical, or imprecise information.
- _____ 2 The axes of the graph are clearly labeled and include the unit of measurement as appropriate.
- _____ 3 The labels are in mixed-case letters and oriented left to right for normal reading.

- ___ 4 Minimum and maximum values are shown on each axis, with up to 9 intermediate markings.
- ___ 5 The starting point on an axis is 0, with the gradations indicated in whole numbers.
- ___ 6 Gradations are at standard intervals; intervening gradations are consistent with the labeled scale interval.
- ___ 7 Labels are used instead of legends or keys to identify the data plotted on the graph.
- ___ 8 Labels are oriented horizontally and located next to the data being referenced.
- ___ 9 Each line or curve on a graph is labeled and coded; critical or abnormal data is coded.
- ___ 10 Grid lines are unobtrusive and do not obscure the data presented in the graph.
- ___ 11 Users can display or suppress grid lines as desired.
- ___ 12 A line graph contains no more than five lines/curves, and each is labeled.
- ___ 13 If data are presented in multiple graphs, the same coding scheme is used in each graph.
- ___ 14 Coding is used to highlight important information, distinguish actual from projected data.
- ___ 15 Multiple trend lines that have to be compared are presented on a single graph.
- ___ 16 Users can redraw multiple graphs using the same scale to facilitate comparison.
- ___ 17 When reading exact values is required, users can display them on the graph and can zoom, and the application provides aids to interpret scale.
- ___ 18 The area under each line in a surface chart is coded and identified by a label displayed in the area.
- ___ 19 If a surface chart is used, the data categories are ordered logically; if no a priori order exists, least variable data categories are on the bottom, most variable on top.

9.2.2 Bar Charts and Histograms

- ___ 1 Bar charts are used to compare a single measure at several intervals, histograms at a large number of intervals.
- ___ 2 Bar charts have a consistent orientation; comparable bars are adjacent.
- ___ 3 Frequency counts are displayed in vertical bars, time durations in horizontal bars.
- ___ 4 A reference index is provided when displayed data are compared with a critical value.
- ___ 5 Bars in a bar chart are separated, using one-half or less of bar width as spacing.
- ___ 6 Coding is used to distinguish among groups of bars, highlight important data in bars.
- ___ 7 In multiple bar charts or histograms, related groups of bars are presented in a consistent order.
- ___ 8 Each bar is identified with its own text label, rather than using a separate legend.
- ___ 9 The design of a bar chart or histogram conforms to user expectations.
- ___ 10 Icons are not used to represent quantitative information.
- ___ 11 Charts and axes are clearly labeled, and important information is highlighted.
- ___ 12 When bars are presented in pairs, they are labeled as a unit and include a legend to distinguish between them.
- ___ 13 Stacked bars are used when both the total measures and the portions represented by segments are of interest.
- ___ 14 If stacked bars are used, the data categories are presented in the same sequence.
- ___ 15 Data categories are ordered with least variable at bottom of bar, most variable at top.
- ___ 16 Areas within each bar are coded and identified by a text label displayed in the area.

9.2.3 Flow Charts

- ___ 1 Flow charts are used to provide a schematic representation of sequences or processes.
- ___ 2 The path indicated in the flow chart is left to right, top to bottom, or clockwise.
- ___ 3 Each decision point in the flow chart contains a single, simple decision.
- ___ 4 The flow chart elements and lines are consistently coded throughout the flowchart.
- ___ 5 The flow chart provides directional indicators to indicate the sequence to be followed.
- ___ 6 A legend describes each element and code; critical information and/or steps are highlighted.
- ___ 7 The steps in the flow chart are ordered logically, by importance, or by certainty; if there is no inherent logic, the steps are ordered to minimize the length of the path through it.
- ___ 8 The shapes used in the flow chart follow existing shape coding conventions.
- ___ 9 The text presented in the chart is oriented for normal reading.
- ___ 10 Important elements (e.g., paths) are emphasized through coding.

9.2.4 Pie Charts

- ___ 1 Pie chart is used to provide an approximation of how an entity is apportioned into parts.
- ___ 2 Each segment is coded in different colors/shading/texture and identified by text label.
- ___ 3 If segment is too small, the label is placed outside, with a line from it to the segment.
- ___ 4 The label describes content of segment and includes number being represented by segment.
- ___ 5 Segments are emphasized by special shading, displacing them from rest of chart.

9.3 TACTICAL INFORMATION

9.3.1 Tactical Symbolology

- ___ 1 Tactical symbolology conforms with available military standards and other relevant national and international standards as required.
- ___ 2 New symbolology created by the application is consistent with applicable standards and matches user expectations.

9.3.2 Coding of Tactical Information

- ___ 1 If color is used to convey tactical meaning, it is used as a redundant code and not as the sole basis for coding.
- ___ 2 Each color represents one category of tactical data.
- ___ 3 Color coding of threat status follows applicable military standards.
- ___ 4 When coding system status, green = operational; yellow = caution; red = inoperative.
- ___ 5 If color is used to show a change in system status, the change is signaled by changing the color of an object (e.g, a box or circle) next to the text.
- ___ 6 Colors selected for tactical coding are used only for that function; if one of these colors is assigned another meaning, a different shade is selected to minimize possible confusion.
- ___ 7 If color is used for alerting, it is assigned only to the information to which attention is directed.

- ___ 8 The standard meaning in terms of alert criticality is assigned to each color; the color has only this meaning.
- ___ 9 Alerting is indicated by assigning color to text information or by adding colored icons to text.

9.4 INFORMATION CODING

9.4.1 Color

- ___ 1 Color is used redundantly and only to provide required functionality; other coding methods are used whenever possible.
- ___ 2 The number of colors used to code information in an alphanumeric display does not exceed 7, and only 4 codes are displayed at any one time.
- ___ 3 The number of colors used to code information in a graphical display does not exceed 8-9.
- ___ 4 When information is color coded, users have the option of displaying the meaning of the code as a reminder.
- ___ 5 If the application uses shading, the colors differ sufficiently in intensity to be easily discriminable.

9.4.2 Flashing

- ___ 1 Flash coding is used only to display urgent information for user attention.
- ___ 2 No more than two levels of flash coding are used.
- ___ 3 The flash rate is 3-5 Hz with equal on/off times; if two levels of flashing are used, the second is 1-2 Hz with equal on/off times.
- ___ 4 For flash coding of a displayed item, a flashing symbol is used; the text does not flash.
- ___ 5 Users acknowledge the event causing the flashing and can suppress it if desired.
- ___ 6 Windows: If the application flashes the title bar of a window, the flashing is accompanied by an auditory signal as a redundant cue.

9.4.3 Reverse Video

- ___ 1 Reverse video is not used for coding in the application.

9.4.4 Size and Shape

- ___ 1 The number of size codes is 5 or less; users have to interpret relative, not absolute size.
- ___ 2 The number of shape codes is limited to 10-20 and relate to the object or operation represented.
- ___ 3 The color and detail added to shapes are the minimum needed to identify meaning of the shape.

9.4.5 Sound

- ___ 1 Auditory signals are used to alert to critical conditions or operations.

- _____ 2 For noncritical auditory alarms, a simple user action acknowledges and turns off the signal.
- _____ 3 Auditory signals are intermittent in nature and allow sufficient time to respond.
- _____ 4 Auditory signals are distinctive in intensity and pitch and do not exceed 4 different levels.
- _____ 5 Signal intensity, duration, and location are appropriate to the environment and personnel.

9.4.6 Text Font and Styles

- _____ 1 No more than two styles of type or two weights are used at one time.
- _____ 2 Variations in type sizes are limited to no more than three at any one time.
- _____ 3 Capitalization is not the sole indication of critical information in a window.
- _____ 4 Underlining is used sparingly and does not conflict with hypermedia conventions.

9.5 DYNAMIC INFORMATION

- _____ 1 Users can control the rate at which dynamically changing information is updated.
- _____ 2 Users can freeze an updated display, then resume at the stoppage or the current time.
- _____ 3 When reading dynamically changing information, the update rate is no more than once per second.
- _____ 4 When identifying rate of change or reading gross values, the update rate is 2-5 times per second.
- _____ 5 Users are prompted to return to automatic updating after freezing a dynamic window.
- _____ 6 Users are informed if significant changes in data occurred while the display was frozen.
- _____ 7 Users are able to control the playback of auditory information (e.g., start, stop, pause) and adjust the volume of the playback.

10.0 TASK-SPECIFIC WINDOW DESIGN

10.1 DATA ENTRY WINDOWS

- _____ 1 Data fields are organized by sequence of use, frequency of use, or importance, with related fields grouped together.
- _____ 2 When users work from a hardcopy form, the window format has an identical format.
- _____ 3 If the window contains different kinds of data fields, they are arranged for efficient data entry using the pointing device or keyboard and to minimize hand movement between input devices.
- _____ 4 For tabular data, entry areas are arranged in rows and columns, with each one labeled.
- _____ 5 In a group of related fields, the labels and text fields are left justified, or the labels are right justified and the text field left justified.
- _____ 6 A conditional field is placed to the right or below the field to which it relates; the field is either shown as unavailable or not displayed until the related control is selected.
- _____ 7 Users can obtain information (e.g., in the message bar, a help window) about a data field and its contents.

10.2 TABULAR DATA WINDOWS

- _____ 1 A tabular data window includes vertical/horizontal scroll bars if the data exceed the space available.
- _____ 2 If the data can be scrolled horizontally, the column heading scrolls with its associated column. If the data can be scrolled vertically, the column heading is outside scrolling area and remains visible and the window includes push buttons for paging.
- _____ 3 When users page through the data, the last line on one page is the first line on the next page.
- _____ 4 The content of the window does not extend over more than one page horizontally.
- _____ 5 If the data in the window can be sorted, users do so by clicking on the column heading.
- _____ 6 The heading remains highlighted after being selected to indicate the column that was sorted.
- _____ 7 If additional sort variations are needed, they are provided in pull-down menus or push buttons in the window.

10.3 LIST-TO-LIST TRANSFER WINDOWS

- _____ 1 A list-to-list transfer window contains source and destination lists and push buttons to transfer between lists.
- _____ 2 The push buttons contain text labels or arrows indicating the direction of the transfer.
- _____ 3 Push buttons are available/unavailable based on the direction of transfer between lists.
- _____ 4 The window can include radio/check buttons or option menus to modify the contents of the source list.
- _____ 5 Users can transfer multiple items but not multiple instances of item to destination list.
- _____ 6 An item in the source list can be copied or moved when transferred to the destination list.
- _____ 7 If transfer is a copy, the item is marked when transferred and unmarked when transferred back.

10.4 MAP WINDOWS

10.4.1 Map Information

- _____ 1 A map window includes identifying information about the map and status information such as coordinates, area, and scale.
- _____ 2 Identifying/status information appears in the message bar or in a subarea of the window itself.
- _____ 3 A continuous indicator of pointer location on the map is available in a standard window area.
- _____ 4 Maps are displayed using the same orientation, and the important features are labeled.
- _____ 5 Labels are positioned consistently with respect to features, do not obscure the features, and remain legible at all display resolutions.
- _____ 6 Map controls appear in the map window or are available in separate dialog windows.
- _____ 7 Users can pan and zoom a map as desired.

- ___ 8 Position or change indicators are provided to return quickly to the normal or starting map.
- ___ 9 Users can define a baseline position on a map and return to this position quickly.
- ___ 10 Users can determine distance and bearing between points and access other functions (e.g., areal computation/verification).
- ___ 11 Users can enter latitude and longitude to the level of accuracy needed.
- ___ 12 Calculations (e.g., range, bearing, position) reflect accuracy appropriate to the scale of the map displayed.

10.4.2 Map Objects

- ___ 1 Map objects are placed accurately or connected to the desired location with arrows, lines, or graphics.
- ___ 2 The label for a map object appears next to the object and presents essential information about it.
- ___ 3 The background of the object and label is transparent so as not to obscure other information.
- ___ 4 The intensity of the map is adjustable so that selected portions of the map can be faded without losing all map features.
- ___ 5 If multiple sets of map objects are available, users can switch between sets without losing data.
- ___ 6 Users can manipulate map objects and change the appearance of information about these objects.
- ___ 7 When a map is zoomed, the size of map objects (including labels) is adjusted to be readable.
- ___ 8 Objects such as overlays include visual indications defining the parts that are selectable.
- ___ 9 The pointing device selection methods in table 10-1 are used to select and deselect map objects.
- ___ 10 Keyboard methods for selecting map objects conform to methods in table 3-2 or 3-3.
- ___ 11 Users can view or declutter overlapping map objects and obtain additional information for selected objects.
- ___ 12 Users can distinguish coincident point objects and obtain information to resolve ambiguities.
- ___ 13 When displaying color overlays, a color coding key is also provided.
- ___ 14 Users can display the coding key as desired without having to redisplay the overlay.
- ___ 15 If the coding key is displayed in a dialog window, it is minimum size and obscures little of overlay.
- ___ 16 If the overlay uses shading, the coding key is a scale so users can interpret the coding in the overlay.

10.5 GRAPHICAL SCHEDULING WINDOWS

10.5.1 Schedule Design

- ___ 1 A graphical scheduling window is used to display timelines or scheduled events.

- _____ 2 A graphical scheduling window has time on the horizontal axis and tasks to be performed arrayed vertically.
- _____ 3 Schedule events are represented by an event icon whose length shows the time needed to complete a task.
- _____ 4 The icon is displayed to the right of its associated task.
- _____ 5 Types of events are indicated by color/shading, with the designator displayed on/above the icon.
- _____ 6 Users can access a legend or key that describes the coding technique used in the schedule.
- _____ 7 No more than nine uniquely coded event icons are presented on a schedule at one time.
- _____ 8 Each icon is labeled if more than one event icon is used per task.
- _____ 9 Event icon labels are placed along the vertical axis or on/above the timeline.
- _____ 10 Different scheduling attributes are represented by displaying symbols with event icons.
- _____ 11 Symbols are formed from geometric shapes, fill patterns to show different situations.
- _____ 12 Gridlines are available if the schedule is cluttered or users require a high degree of precision.
- _____ 13 The gridline indicates present date and time; users can show or hide the line as needed.

10.5.2 Schedule Manipulation

- _____ 1 Users can define the start and stop time of the schedule to the desired degree of precision.
- _____ 2 Users can display all or part of the preselected duration time.
- _____ 3 Users can select an individual event icon and obtain additional information about the event.
- _____ 4 The pointing device selection methods available to users comply with those in table 3-1.
- _____ 5 Users can reschedule an event icon using transfer methods described in section 3.5.
- _____ 6 If exact positioning is difficult, users have alternative methods to locate the icon.

11.0 USER SUPPORT RESOURCES

11.1 OBJECT-LEVEL HELP

11.1.1 Message Bar Information

- _____ 1 If a window has a message bar, object-level help is displayed in that area when the object has keyboard focus.

11.1.2 Context-Sensitive Help

- _____ 1 If context-sensitive help is available, <Shift> <Help> in Motif or <Shift><F1> in Windows invokes this mode and changes the pointer to a “help” shape.
- _____ 2 Users place the pointer on a window component and click BSelect to display information about the component in a pop-up Help window.
- _____ 3 The pop-up Help window contains a brief description of the component and how to use it.

- _____ 4 Motif: In windows with a menu bar, access to context-sensitive help is available by selecting the On Item option from the Help menu and clicking BSelect on a window component.
- _____ 5 Help information is provided whenever users invoke context-sensitive help; users do not receive a “Help not available” message.

11.2 WINDOW-LEVEL HELP

11.2.1 Window Design

- _____ 1 Window-level help is available for every window in the application.
- _____ 2 Users access this help by activating a Help push button or menu option or by pressing <F1> (or <Help> in Motif) in an application window.
- _____ 3 The Help button or option is dimmed and unavailable for selection if help is not available; an empty help window (or one indicating that no help is available) is not displayed.
- _____ 4 The title of a help window includes the name of the application window for which help was requested.
- _____ 5 The window is large enough to display at least ten lines of text in the display area and wide enough to display an entire line of text.
- _____ 6 The window includes scroll bars if the text exceeds the available display area.
- _____ 7 The window has an OK push button which is the default in the window and may include Previous, Next, and More push buttons.
- _____ 8 When a help window appears, it is located to the right/left/above/below, but does not cover the component for which help was requested.
- _____ 9 When a help window appears, it displays information at beginning of the help description.
- _____ 10 A help window can be moved and resized and is modeless.
- _____ 11 Users can print any help window by selecting all or marking the beginning and end of part of the text.
- _____ 12 A help window is removed when the parent application window is minimized or closed.
- _____ 13 When users close a help window, focus returns to the application window for which help was requested.

11.2.2 Window Content

- _____ 1 The help window presents only the information related to the application window for which it provides support.
- _____ 2 The help window provides information on the purpose of the window and the actions available.
- _____ 3 Help can explain procedures for performing the task(s) presented in the window or include a More push button that provides access to this information.
- _____ 4 Text is bulleted, steps are numbered, and explanations are presented in columns.
- _____ 5 Graphics are included only if essential to understand task in the application window.
- _____ 6 When presenting a sequence of steps, the explanation follows the same sequence.

11.3 APPLICATION-LEVEL HELP (TBD)

11.4 SYSTEM-LEVEL HELP

- _____ 1 Motif: Users can access system-level support resources from Help Manager as well as obtain help on how to use the desktop and the Front Panel.